

FFFFFF	000000000	RRRRRRRRRRRR	RRRRRRRRRRRR	TTTTTTTTTTTTT	LLL	
FFFF	000000000	RRRRRRRRRRRR	RRRRRRRRRRRR	TTTTTTTTTTTTT	LLL	
FFFF	000000000	RRRRRRRRRRRR	RRRRRRRRRRRR	TTTTTTTTTTTTT	LLL	
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFFF	000	000	RRRRRRRRRRRR	RRRRRRRRRRRR	TTT	LLL
FFFF	000	000	RRRRRRRRRRRR	RRRRRRRRRRRR	TTT	LLL
FFFF	000	000	RRRRRRRRRRRR	RRRRRRRRRRRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000	000	RRR	RRR	TTT	LLL
FFF	000000000	RRR	RRR	RRR	TTT	LLL
FFF	000000000	RRR	RRR	RRR	TTT	LLL
FFF	000000000	RRR	RRR	RRR	TTT	LLL

\*\*FILE\*\*ID\*\*FORLIB

M 9

0001 0 FORTRAN Run-Time Library specific macros and symbols  
0002 0 File: FORLIB.REQ, Edit: SBL1003  
0003 0  
0004 0 \*\*\*\*\*  
0005 0 \*  
0006 0 \* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
0007 0 \* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
0008 0 \* ALL RIGHTS RESERVED.  
0009 0 \*  
0010 0 \* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
0011 0 \* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
0012 0 \* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
0013 0 \* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
0014 0 \* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
0015 0 \* TRANSFERRED.  
0016 0 \*  
0017 0 \* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
0018 0 \* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
0019 0 \* CORPORATION.  
0020 0 \*  
0021 0 \* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
0022 0 \* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  
0023 0 \*  
0024 0 \*  
0025 0 \*\*\*\*\*  
0026 0  
0027 0 Author: Steven B. Lionel, 23-September-1982  
0028 0  
0029 0 1-001 - Original. SBL 23-September-1982  
0030 0 1-002 - Add FORMACROS.REQ. 1-Mar-1983  
0031 0 1-003 - Add FORRCE.R32. SBL 2-Jun-1983  
0032 0 --  
0033 0  
0034 0 +  
0035 0 This file is the master source for FORLIB.L32.  
0036 0 It contains definitions for macros and symbols used internally to the  
0037 0 FORTRAN Run-Time Library.  
0038 0 -  
0039 0  
0040 0 SWITCHES ADDRESSING\_MODE (EXTERNAL=GENERAL, NONEXTERNAL=WORD\_RELATIVE);  
0041 0  
0042 0 LIBRARY 'RTLSTARLE'; ! SYSSLIBRARY:STARLET.L32  
0043 0  
0044 0 REQUIRE 'RTLML:FORERR'; ! FOR\$K\_error codes

```

R0045 0 ! **** MODULE $FORERR ***
R0046 0 ! Created 15-SEP-1984 22:45:50 by VAX-11 SDL V2.0      Source: 15-SEP-1984 22:44:40 $255$DUA28:[FORRTL.SRC]FOR
R0047 0 ! ****
R0048 0
R0049 0
R0050 0 !*** MODULE $FORERR ***
R0051 0 literal FOR$K_FAC NO = 24;
R0052 0 literal FOR$K_NOTFORSPE = 1;
R0053 0 literal FOR$K_SYNERRNAM = 17;
R0054 0 literal FOR$K_TOOMANVAL = 18;
R0055 0 literal FOR$K_INVREFVAR = 19;
R0056 0 literal FOR$K_REWERR = 20;
R0057 0 literal FOR$K_DUPFILSPE = 21;
R0058 0 literal FOR$K_INPRECTOO = 22;
R0059 0 literal FOR$K_BACERR = 23;
R0060 0 literal FOR$K_ENDDURREA = 24;
R0061 0 literal FOR$K_RECNUMOUT = 25;
R0062 0 literal FOR$K_OPEDEFREQ = 26;
R0063 0 literal FOR$K_TOOMANREC = 27;
R0064 0 literal FOR$K_CLOERR = 28;
R0065 0 literal FOR$K_FILNOTFOU = 29;
R0066 0 literal FOR$K_OPEFAI = 30;
R0067 0 literal FOR$K_MIXFILACC = 31;
R0068 0 literal FOR$K_INVLOGUNI = 32;
R0069 0 literal FOR$K_ENDFILER = 33;
R0070 0 literal FOR$K_UNIALROPE = 34;
R0071 0 literal FOR$K_SEGRECFOR = 35;
R0072 0 literal FOR$K_ATTACNON = 36;
R0073 0 literal FOR$K_INCRECLEN = 37;
R0074 0 literal FOR$K_ERRDURWRI = 38;
R0075 0 literal FOR$K_ERRDURREA = 39;
R0076 0 literal FOR$K_RECIO_OPE = 40;
R0077 0 literal FOR$K_INSVIRMEM = 41;
R0078 0 literal FOR$K_NO_SUCDEV = 42;
R0079 0 literal FOR$K_FICNAMSPE = 43;
R0080 0 literal FOR$K_INRECTYP = 44;
R0081 0 literal FOR$K_KEYVALERR = 45;
R0082 0 literal FOR$K_INCOPECLO = 46;
R0083 0 literal FOR$K_WIREAFIL = 47;
R0084 0 literal FOR$K_INVARGFOR = 48;
R0085 0 literal FOR$K_INVKYESPE = 49;
R0086 0 literal FOR$K_INCKEYCHG = 50;
R0087 0 literal FOR$K_INCFILORG = 51;
R0088 0 literal FOR$K_SPERECLOC = 52;
R0089 0 literal FOR$K_NO_CURREC = 53;
R0090 0 literal FOR$K_REWRITER = 54;
R0091 0 literal FOR$K_DELERR = 55;
R0092 0 literal FOR$K_UNLERR = 56;
R0093 0 literal FOR$K_FINERR = 57;
R0094 0 literal FOR$K_LISIO_SYN = 59;
R0095 0 literal FOR$K_INFFORLOO = 60;
R0096 0 literal FOR$K_FORVARMIS = 61;
R0097 0 literal FOR$K_SYNERRFOR = 62;
R0098 0 literal FOR$K_OUTCONERR = 63;
R0099 0 literal FOR$K_INPCONERR = 64;
R0100 0 literal FOR$K_OUTSTAOVE = 66;
R0101 0 literal FOR$K_INPSTAREQ = 67;

```

15-Sep-1984 23:44:38  
15-Sep-1984 22:45:53

VAX-11 Bliss-32 V4.0-742  
\$255\$DUA28:[FORRTL.OBJ]FORERR.R32;1 Page 3 (1)

; R0102 0 literal FOR\$K\_VFEVALERR = 68;  
; R0103 0 literal FOR\$K\_INTOVF = 70;  
; R0104 0 literal FOR\$K\_INTZERDIV = 71;  
; R0105 0 literal FOR\$K\_FLOOVE = 72;  
; R0106 0 literal FOR\$K\_FLOZERDIV = 73;  
; R0107 0 literal FOR\$K\_FLOUND = 74;  
; R0108 0 literal FOR\$K\_DECSTROVE = 76;  
; R0109 0 literal FOR\$K\_ARRREFOUT = 77;  
; R0110 0 literal FOR\$K\_ADJARRDIM = 93;  
; R0111 0 literal FOR\$K\_MAX\_ERR = 93;

D 10  
15-Sep-1984 23:44:38  
15-Sep-1984 22:44:59

VAX-11 Bliss-32 V4.0-742  
\$255\$DUA28:[FORRTL.SRC]FORLIB.REQ;1 Page 4 (1)

: 0112 0  
: 0113 0 REQUIRE 'RTLIN:FORFMT';

! FORMAT codes and fields

File: FORFMT.REQ Edit: JAW1004

R0114 0  
 R0115 0  
 R0116 0 This file, FORFMT.REQ, defines symbols for the VAX-11 FORTRAN  
 R0117 0 formatting routines.  
 R0118 0  
 R0119 0  
 R0120 0 \*\*\*\*\*  
 R0121 0 \*  
 R0122 0 \* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
 R0123 0 \* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
 R0124 0 \* ALL RIGHTS RESERVED.  
 R0125 0 \*  
 R0126 0 \* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 R0127 0 \* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 R0128 0 \* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 R0129 0 \* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 R0130 0 \* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 R0131 0 \* TRANSFERRED.  
 R0132 0 \*  
 R0133 0 \* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 R0134 0 \* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 R0135 0 \* CORPORATION.  
 R0136 0 \*  
 R0137 0 \* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 R0138 0 \* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  
 R0139 0 \*  
 R0140 0 \*  
 R0141 0 \*\*\*\*\*

## R0145 0 Revision History:

R0147 0 0-12 - Change name to FORFMT.REQ JBS 14-NOV-78  
 R0148 0 1-001 - Add copyright notice and change version number JBS 16-NOV-78  
 R0149 0 1-002 - Add FORTRAN-77 format codes. SBL 09-Feb-1979  
 R0150 0 1-003 - Remove PRINT statement, for new BLISS compiler. JBS 02-OCT-1979  
 R0151 0 1-004 - Add V\_RC\_TYPE\_BYT and V\_RC\_TYPE\_WORD. JAW 10-Aug-1981  
 R0152 0 --

R0153 0  
 R0154 0 \* Define format code byte  
 R0155 0 !-

## R0158 0 MACRO

R0159 0 V\_FMT\_CODE = 0,0,7,0 %; ! 7-bit format code  
 R0160 0 V\_FMT\_REPRE = 0,7,1,0 %; ! Optional representation byte present?

## R0162 0 MACRO

R0163 0 !  
 R0164 0 \* Optional format representation byte:  
 R0165 0 \* The following are bits in the optional format representation:  
 R0166 0 \* byte. The byte is copied into local BLOCK  
 R0167 0 \* B\_FMT\_REPRESENT. These flags indicate less  
 R0168 0 \* frequently used sizes of the format code representations.  
 R0169 0  
 R0170 0

```

R0171 0
R0172 0
R0173 0
R0174 0
R0175 0
R0176 0
R0177 0
R0178 0
R0179 0
R0180 0
R0181 0
R0182 0
R0183 0
R0184 0
R0185 0
R0186 0
R0187 0
R0188 0
R0189 0
R0190 0
R0191 0
R0192 0
R0193 0
R0194 0
R0195 0
R0196 0
R0197 0
R0198 0
R0199 0
R0200 0
R0201 0
R0202 0
R0203 0
R0204 0
R0205 0
R0206 0
R0207 0
R0208 0
R0209 0
R0210 0
R0211 0
R0212 0
R0213 0
R0214 0
R0215 0
R0216 0
R0217 0
R0218 0
R0219 0
R0220 0
R0221 0
R0222 0
R0223 0
R0224 0
R0225 0
R0226 0
R0227 0

V_RC_TYPE = 0,0,2,0 %. | Repeat count type:
V_RC_TYPE_BYT E = 0,0,1,0 %. | 0 = not present, 1 = byte, 2 = word
V_RC_TYPE_WORD = 0,1,1,0 %. |
V_W_WORD = 0,2,1,0 %. | 0=W field is byte, 1=W field is word.
V_E_VFE = 0,4,1,0 %. | Bit 3 is reserved to DEC
V_D_VFE = 0,5,1,0,%. | E field is VFE
V_W_VFE = 0,6,1,0 %. | D field is VFE
V_RC_VFE = 0,7,1,0 %. | W field is a VFE (ignore V_W_WORD)
                           | Repeat count field is a VFE
                           | ignore V_RC_BYT E and V_RC_WORD

+ Define format code symbols which are is 2 or 3 characters so it plus comma will fit
  in between logical tabs. One character symbols are prefixed with so
  that they are two character symbols instead (so won't conflict with LOCALS).

LITERAL
ER      = 0.      | 00      | Format syntax error - only from object time format compiler
LP      = 1.      | 01      | ( - Format reversion point
NLP     = 2.      | 02      | n( - Left paren of repeat group
RP      = 3.      | 03      | ) - Right paren of repeat group
EOF     = 4.      | 04      | ) - End of format
SLS     = 5.      | 05      | / - Record separator
DLR     = 6.      | 06      | $ - Dollar sign: terminal I/O
CLN     = 7.      | 07      | do not return to left margin
                           | : - Colon: terminate if end of list
                           | so no trailing Hollerith printed

S       = 9.      | 09      | S - Restore + optional
SP      = 10.     | 0A      | SP - Force + on
SS      = 11.     | 0B      | SS - Force + off
-P      = 12.     | 0C      | sP - signed scale factor (-128 =< s =< +127).
-T      = 13.     | 0D      | Tn - Tab Set (0 < n =< 32767)
-X      = 14.     | 0E      | nX - Skip n columns (0 < n =< 32767)
-H      = 15.     | 0F      | nHcccc - Hollerith: n chars follow (0 < n =< 32767)
BN      = 16.     | 10      | BN - Blanks are nulls
BZ      = 17.     | 11      | BZ - Blanks are zeroes
TL      = 18.     | 12      | TLC - Tab left c columns
TR      = 19.     | 13      | TRC - Tab right c columns

-Q      = 20.     | 14      | Q - no. of input chars left in record
-A      = 21.     | 15      | nAw - Alpha numeric
-MIN DATA = A.   | 16      | Minimum I/O list transmitting data code
-L      = 22.     | 16      | nLw - Locial
-MIN INT = L.   | 17      | Min. integer
-O      = 23.     | 17      | n0w - Octal
-I      = 24.     | 18      | niw - Integer
-Z      = 25.     | 19      | nzw - Hexadecimal
-X0     = 26.     | 1A      | 0w.m - Extended 0
-X1     = 27.     | 1B      | 1w.m - Extended 1
-XZ     = 28.     | 1C      | 2w.m - Extended 2
-MAX INT = XZ.  |          | Max. integer (not counting defaults)

```

```

R0228 0
R0229 0
R0230 0
R0231 0
R0232 0
R0233 0
R0234 0
R0235 0
R0236 0
R0237 0
R0238 0
R0239 0
R0240 0
R0241 0
R0242 0
R0243 0
R0244 0
R0245 0
R0246 0
R0247 0
R0248 0
R0249 0
R0250 0
R0251 0
R0252 0
R0253 0
R0254 0
R0255 0
R0256 0
R0257 0
R0258 0
R0259 0
R0260 0
R0261 0
R0262 0

: -F      = 30,   ! 1E      ! nFw.d - Fixed format
: -MIN_FLT = 31,   ! 1F      ! Min. floating
: -E      = 32,   ! 20      ! nEw.d - Scientific notation format
: -G      = 33,   ! 21      ! nGw.d - General format
: -D      = 34,   ! 22      ! nDw.d - Double Precision format
: RE     = 35,   ! 23      ! nEw.dEe - Extended E
: XG     = 35,   ! 23      ! nGw.dEe - Extended G
: MAX_FLT = XG, !          ! max. floating (not counting default)
: MAX_DATA = XG, !          ! Max. data (not counting default)

: + Default format codes:
: -
: -DA     = 41,   ! 29      ! nA - default A
: -DL     = 42,   ! 2A      ! nL - default L
: -DO     = 43,   ! 2B      ! nO - default O
: -DI     = 44,   ! 2C      ! nI - default I
: -DZ     = 45,   ! 2D      ! nZ - default Z
: -DF     = 50,   ! 32      ! nF - default F
: -DE     = 51,   ! 33      ! nE - default E
: -DG     = 52,   ! 34      ! nG - default G
: -DD     = 53,   ! 35      ! nD - default D

: + Note: 0 < n <= 32767 (decimal)
:        0 < w <= 65535 (decimal)
:        0 <= d <= 255 (decimal)
:        0 <= e <= 255 (decimal)
: -
: End of file FORFMT.REQ

```

H 10  
15-Sep-1984 23:44:38  
15-Sep-1984 22:44:59

VAX-11 Bliss-32 V4.0-742  
\_S255\$DUA28:[FORRTL.SRC]FORLIB.REQ;1 Page 8 (1)

0263 0  
0264 0 REQUIRE 'RTLIN:FORMACROS';

! FORTRAN-specific macros

R0265 0 Macros for FORTRAN Run-Time Library  
R0266 0 File: FORMACROS.REQ, Edit: SBL1001  
R0267 0 \*\*\*\*\*  
R0268 0 \*  
R0269 0 \* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
R0270 0 \* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
R0271 0 \* ALL RIGHTS RESERVED.  
R0272 0 \*  
R0273 0 \* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
R0274 0 \* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
R0275 0 \* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
R0276 0 \* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
R0277 0 \* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
R0278 0 \* TRANSFERRED.  
R0279 0 \*  
R0280 0 \*  
R0281 0 \* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
R0282 0 \* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
R0283 0 \* CORPORATION.  
R0284 0 \*  
R0285 0 \* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
R0286 0 \* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  
R0287 0 \*  
R0288 0 \*  
R0289 0 \*\*\*\*\*\*  
R0290 0 \*  
R0291 0 Author: Steven B. Lionel, 7-Jan-1983  
R0292 0  
R0293 0 1-001 - Original. SBL 7-Jan-1983  
R0294 0  
R0295 0  
R0296 0  
R0297 0 \* Macro to call FOR\$SIGNAL\_STO and return. This saves lines of code  
R0298 0 \* in the source. It can replace a call to FOR\$SIGNAL\_STO anywhere.  
R0299 0  
R0300 0  
R0301 0  
R0302 0  
MR0303 0 \*  
MR0304 0 \* MACRO  
MR0305 0 \*  
MR0306 0 \*  
MR0307 0 \*  
MR0308 0 \*  
MR0309 0 \*  
MR0310 0 \*  
MR0311 0 \*  
MR0312 0 \*  
R0313 0 \*  
R0314 0 \*  
R0315 0 \*  
R0316 0 \* Structure definitions used to declare the FAB and NAM as being offset from  
R0317 0 \* CCB. To use, make the following declarations:  
R0318 0  
R0319 0  
R0320 0  
R0321 0  
BIND  
FAB = CCB: REF \$FOR\$FAB\_CCB\_STRUCT,  
NAM = CCB: REF \$FOR\$NAM\_CCB\_STRUCT;

10  
15-Sep-1984 23:44:38  
15-Sep-1984 22:45:03

VAX-11 Bliss-32 V4.0-742  
[FORRTL.SRC]FORMACROS.REQ;1

Page 10  
(1)

RO322 0  
RO323 0  
RO324 0  
RO325 0  
RO326 0  
RO327 0  
RO328 0  
RO329 0  
RO330 0  
RO331 0  
RO332 0  
RO333 0  
RO334 0  
|.  
STRUCTURE  
\$FOR\$FAB\_CCB STRUCT [O, P, S, E] =  
[FAB\$C\_B[N]  
(\$FOR\$FAB\_CCB\_STRUCT+RAB\$C\_BLN+0)<P,S,E>.  
\$FOR\$NAM\_CCB STRUCT [O, P, S, E] =  
[NAM\$C\_B[N]  
(\$FOR\$NAM\_CCB\_STRUCT+RAB\$C\_BLN+FAB\$C\_BLN+0)<P,S,E>;  
: End of file FORMACROS.REQ

K 10

15-Sep-1984 23:44:38  
15-Sep-1984 22:44:59

VAX-11 Bliss-32 V4.0-742

\_S255\$DUA28:[FORRTL.SRC]FORLIB.REQ;1

Page 11  
(1)

0335 0  
0336 0

REQUIRE 'RTLML:FORMSG';

! FOR\$\_ error codes

R0337 0  
R0338 0  
R0339 0  
R0340 0  
R0341 0  
R0342 0  
R0343 0  
R0344 0  
R0345 0  
R0346 0  
R0347 0  
R0348 0  
R0349 0  
R0350 0  
R0351 0  
R0352 0  
R0353 0  
R0354 0  
R0355 0  
R0356 0  
R0357 0  
R0358 0  
R0359 0  
R0360 0  
R0361 0  
R0362 0  
R0363 0  
R0364 0  
R0365 0  
R0366 0  
R0367 0  
R0368 0  
R0369 0  
R0370 0  
R0371 0  
R0372 0  
R0373 0  
R0374 0  
R0375 0  
R0376 0  
R0377 0  
R0378 0  
R0379 0  
R0380 0  
R0381 0  
R0382 0  
R0383 0  
R0384 0  
R0385 0  
R0386 0  
R0387 0  
R0388 0  
R0389 0  
R0390 0  
R0391 0  
R0392 0  
R0393 0

\*\*\*\*\*  
[Created 15-SEP-1984 22:46:32 by VAX-11 SDL V2.0      Source: 15-SEP-1984 22:45:39 \$255\$DUA28:[FORRTL.OBJ]FORMSG.R32;1]  
\*\*\*\*\*  
\*\*\* MODULE \$FORDEF \*\*\*  
This SDL File Generated by VAX-11 Message V04-00 on 15-SEP-1984 22:45:40.89  
FILE: FORMSG.MSG EDIT: SBL2006  
\*\*\*\*\*  
\*  
\*      COPYRIGHT (C) 1978, 1980, 1982, 1984 BY  
\*      DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
\*      ALL RIGHTS RESERVED.  
\*  
\*      THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
\*      ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
\*      INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
\*      COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
\*      OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
\*      TRANSFERRED.  
\*  
\*      THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
\*      AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
\*      CORPORATION.  
\*  
\*      DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
\*      SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  
\*  
\*\*\*\*\*  
1-014 - ADD ERRORS 17-19 FOR NAMELIST. SBL 10-NOV-1980  
1-015 - CORRECT TYPO IN TOOMANVAL. SBL 15-DEC-1980  
1-016 - RESTORE CLOSING BRACKET ON OPEDEFREQ, MYSTERIOUSLY DROPPED IN 1-015.  
SBL 6-JAN-1981  
2-001 - CONVERT TO MESSAGE UTILITY SOURCE FORMAT. ADAPTED FROM FORMSG.MDL  
VERSION 1-016. SBL 22-APR-1981  
2-002 - ADD UNFI0\_FMT, FMTIO\_UNF, DIRIO\_KEY, SEQIO\_DIR, KEYIO\_DIR,  
IO\_NONFOR, -INVXTREC, -FLOUNDEXC. JAW 23-AUG-1981  
2-003 - CHANGE INVXTREC TO INVTEXREC. JAW 24-AUG-1981  
2-004 - ADD INVTEX WHICH DOES NOT REQUIRE A RECORD NUMBER. THIS IS FOR INDEXED  
AND INTERNAL FILES THAT DO NOT HAVE A VALID RECORD NUMBER. DGP 21-DEC-9181  
2-005 - ADD .TITLE. SBL 28-AUG-1982  
2-006 - ADD OPEREQDIS AND OPEREQSEQ. SBL 2-JUN-1983  
--  
\*  
SYMBOLS ARE DEFINED FOLLOWING THE STANDARD FOR GLOBAL NAMES:  
FOR\$\_ABCMNOXYZ  
IN ADDITION, THE LIBRARY STANDARDS SPECIFY THAT THE LETTERS "abc", "mno",  
AND "xyz" ARE THE FIRST THREE LETTERS OF THE FIRST THREE WORDS OF THE ERROR  
MESSAGE, NOT COUNTING ARTICLES AND PREPOSITIONS.

M 10  
15-Sep-1984 23:44:38  
15-Sep-1984 22:46:38

VAX-11 Bliss-32 V4.0-742  
S255SDUA28:[FORRTL.OBJ]FORMSG.R32;1

Page 13  
(1)

R0394 0  
R0395 0  
R0396 0  
R0397 0  
R0398 0  
R0399 0  
R0400 0  
R0401 0  
R0402 0  
R0403 0  
R0404 0  
R0405 0  
R0406 0  
R0407 0  
R0408 0  
R0409 0  
R0410 0  
R0411 0  
R0412 0  
R0413 0  
R0414 0  
R0415 0  
R0416 0  
R0417 0  
R0418 0  
R0419 0  
R0420 0  
R0421 0  
R0422 0  
R0423 0  
R0424 0  
R0425 0  
R0426 0  
R0427 0  
R0428 0  
R0429 0  
R0430 0  
R0431 0  
R0432 0  
R0433 0  
R0434 0  
R0435 0  
R0436 0  
R0437 0  
R0438 0  
R0439 0  
R0440 0  
R0441 0  
R0442 0  
R0443 0  
R0444 0  
R0445 0  
R0446 0  
R0447 0  
R0448 0  
R0449 0  
R0450 0

THE NAMES OF ALL PROCEDURES USING EACH ERROR CONDITION VALUE SHOULD APPEAR IN THE COMMENTS INCLUDED WITH EACH ERROR DEFINITION.

MACRO-32 PROGRAMMING:

THE MACROS CALL:

\$FORDEF

WILL CAUSE ALL SYMBOLS TO BE MADE AVAILABLE TO THE MODULE.  
THE STSSV MSG\_ID IS THE FORTRAN ERROR NUMBER (1:94).  
THESE SYMBOLS ARE DECLARED EXTERNAL BY THE RUN-TIME LIBRARY.  
THE MACROS ARE PROVIDED ONLY FOR THE CONVENIENCE OF THE USER.

: MAKE ALL ERRORS BE SEVERE (EXCEPT AS NOTED)  
: SET SUB-SYSTEM SPECIFIC BIT AND MAKE SEVERE  
: SET LH TO 24 (DECIMAL).

literal FORS\_FACILITY = 24:  
literal FORS\_NOTFORSPE = 1605644:  
: NEVER SIGNALLED. USED ONLY AS A FORTRAN ERROR  
: NUMBER FOR ERRNSNS TO MEAN SOME OTHER FACILITY  
: THAN FORS OR MTH\$ DETECTED THE ERROR.  
: SET SUB-SYSTEM SPECIFIC BIT AND MAKE SEVERE  
: SET LH TO 24 (DECIMAL).

literal FORS\_SYNERRNAM = 1605772:  
literal FORS\_TOOMANVAL = 1605780:  
literal FORS\_INVREFVAR = 1605788:  
literal FORS\_REWERR = 1605796:  
literal FORS\_DUPFILESP = 1605804:  
literal FORS\_INPRECTOO = 1605812:  
literal FORS\_BACERR = 1605820:  
literal FORS\_ENDDURREA = 1605828:  
literal FORS\_RECNUMOUT = 1605836:  
literal FORS\_OPEDEFREQ = 1605844:  
literal FORS\_TOOMANREC = 1605852:  
literal FORS\_CLOERR = 1605860:  
literal FORS\_FILNOTFOU = 1605868:  
literal FORS\_OPEFAI = 1605876:  
literal FORS\_MIXFILACC = 1605884:  
literal FORS\_INVLOGUNI = 1605892:  
literal FORS\_ENDFILEERR = 1605900:  
literal FORS\_UNIALROPE = 1605908:  
literal FORS\_SEGRECFOR = 1605916:  
literal FORS\_ATTACCNON = 1605924:  
literal FORS\_INCRECLEN = 1605932:  
literal FORS\_ERRDURWRI = 1605940:  
literal FORS\_ERRDURREA = 1605948:  
literal FORS\_RECIO\_OPE = 1605956:  
literal FORS\_INSVIRMEM = 1605964:  
literal FORS\_NO\_SUCDEV = 1605972:  
literal FORS\_FICNAMSPE = 1605980:  
literal FORS\_INCRECTYP = 1605988:  
literal FORS\_KEYVALERR = 1605996:  
literal FORS\_INCOPECLO = 1606004:  
literal FORS\_WRIREADIL = 1606012:  
literal FORS\_INVARGFOR = 1606020:

```

R0451 0 literal FORS_INVKEYSPE = 1606028;
R0452 0 literal FORS_INCKEYCHG = 1606036;
R0453 0 literal FORS_INCFILORG = 1606044;
R0454 0 literal FORS_SPERECLOC = 1606052;
R0455 0 literal FORS_NO_CURREC = 1606060;
R0456 0 literal FORS_REREITER = 1606068;
R0457 0 literal FORS_DELERR = 1606076;
R0458 0 literal FORS_UNLERR = 1606084;
R0459 0 literal FORS_FINERR = 1606092;
R0460 0 literal FORS_MORONEREC = 1605852;
R0461 0 literal FORS_ATTREANON = 1605924;
R0462 0 ! SET SUB-SYSTEM SPECIFIC BIT AND MAKE SEVERE
R0463 0 literal FORS_LISIO_SYN = 1606108;
R0464 0 literal FORS_INFFORL00 = 1606116;
R0465 0 literal FORS_FORVARMIS = 1606124;
R0466 0 literal FORS_SYNERRFOR = 1606132;
R0467 0 ! SET SUB-SYSTEM SPECIFIC BIT AND MAKE ERROR (NOT SEVERE)
R0468 0 literal FORS_OUTCONERR = 1606138;
R0469 0 ! SET SUB-SYSTEM SPECIFIC BIT AND MAKE SEVERE
R0470 0 literal FORS_INPCONERR = 1606148;
R0471 0 ! SET SUB-SYSTEM SPECIFIC BIT AND MAKE SEVERE
R0472 0 literal FORS_OUTSTAOVE = 1606164;
R0473 0 literal FORS_INPSTAREQ = 1606172;
R0474 0 literal FORS_VFEVALERR = 1606180;
R0475 0 ! SET SUB-SYSTEM SPECIFIC BIT AND MAKE SEVERE
R0476 0 literal FORS_ADJARRDIM = 1606380;
R0477 0 ! *****
R0478 0 THE FOLLOWING MESSAGES ARE SECONDARY MESSAGES, OR ARE USED ONLY IN
R0479 0 EXIT HANDLERS, AND THEREFORE DO NOT NEED TO BE (AND SHOULD NOT BE)
R0480 0 CONTIGUOUS WITH THE MESSAGES ABOVE.
R0481 0 THE NEXT SIX MESSAGES ARE SECONDARY MESSAGES FOR USE WITH MIXFILACC IN
R0482 0 FORSSIO BEG AND FORSSCB.
R0483 0 literal FORS_UNFI0_FMT = 1607684;
R0484 0 literal FORS_FMTI0_UNF = 1607692;
R0485 0 literal FORS_DIRIO_KEY = 1607700;
R0486 0 literal FORS_SEQIO_DIR = 1607708;
R0487 0 literal FORS_KEYIO_DIR = 1607716;
R0488 0 literal FORS_IO_NORFOR = 1607724;
R0489 0 ! THE NEXT TWO MESSAGES ARE SECONDARY MESSAGES FOR USE WITH INPCONERR IN
R0490 0 FORSSUDF WF AND FORSSUDF WL;
R0491 0 literal FORS_INVTEXREC = 1607732;
R0492 0 literal FORS_INVTEX = 1607740;
R0493 0 ! ADDITIONAL SECONDARY MESSAGES
R0494 0 literal FORS_OPEREQDIS = 1607748;
R0495 0 literal FORS_OPEREQSEQ = 1607756;
R0496 0 ! THE NEXT MESSAGE IS A PRIMARY MESSAGE USED IN THE EXIT HANDLER WHICH
R0497 0 IS DECLARED BY FORSHANDLER.
R0498 0 literal FORS_FLOUNDEXC = 1608035;
R0499 0 ! END OF SPECIAL MESSAGES.
R0500 0 ! *****

```

8 11  
15-Sep-1984 23:44:38  
15-Sep-1984 22:44:59

VAX-11 Bliss-32 V4.0-742  
S255\$DUA28:[FORRTL.SRC]FORLIB.REQ;1 Page 15 (1)

0501 0  
0502 0    REQUIRE 'RTLIN:FORNML';

! NAMELIST definitions

```

R0503 0 ! FORNML.REQ - NAMELIST NMLS Definitions - Version 1-002 - Edit: SBL1002
R0504 0
R0505 0
R0506 0
R0507 0
R0508 0
R0509 0
R0510 0
R0511 0
R0512 0
R0513 0
R0514 0
R0515 0
R0516 0
R0517 0
R0518 0
R0519 0
R0520 0
R0521 0
R0522 0
R0523 0
R0524 0
R0525 0
R0526 0
R0527 0
R0528 0
R0529 0
R0530 0
R0531 0
R0532 0
R0533 0
R0534 0
R0535 0
R0536 0
R0537 0
R0538 0
R0539 0
R0540 0
R0541 0
R0542 0
R0543 0
R0544 0
R0545 0
R0546 0
R0547 0
R0548 0
R0549 0
R0550 0
R0551 0
R0552 0
R0553 0
R0554 0
R0555 0
R0556 0
R0557 0
R0558 0
R0559 0

*****  

* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  

* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  

* ALL RIGHTS RESERVED.  

*  

* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  

* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  

* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  

* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  

* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  

* TRANSFERRED.  

*  

* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  

* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  

* CORPORATION.  

*  

* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  

* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  

*  

*****  

* AUTHOR: Steven B. Lionel  

*  

* EDIT HISTORY:  

* 1-001 - Original. SBL 21-August-1980  

* 1-002 - Add NMLSV_SUBSCRIPT. SBL 15-April-1981  

*  

* LITERAL  

* NMLSK_BLKLENGTH = TPASK_LENGTH0 + 120; ! TPARSE parameter block length  

*  

* FIELD  

* NMLSFIELDS =  

* +  

* The following fields are part of the TPARSE parameter block past that  

* used by LIBSTPARSE.  

* -  

* SET  

* NMLSA_LISTBLOCK = [TPASK_LENGTH0 + 00. 0. 32. 0]. NAMELIST descriptor block  

* NMLSA_VARNAME = [TPASK_LENGTH0 + 04. 0. 32. 0]. Variable name  

* NMLSA_VARSTART = [TPASK_LENGTH0 + 08. 0. 32. 0]. Variable start address  

* NMLSA_VAREND = [TPASK_LENGTH0 + 12. 0. 32. 0]. Variable end address  

* NMLSA_VARCUR = [TPASK_LENGTH0 + 16. 0. 32. 0]. Current position  

* NMLSW_VARSIZE = [TPASK_LENGTH0 + 20. 0. 16. 0]. Segment size  

* NMLSW_STRIDE = [TPASK_LENGTH0 + 22. 0. 16. 0]. Stride between elements  

* NMLSA_DESCR = [TPASK_LENGTH0 + 24. 0. 32. 0]. Address of descriptor  

* NMLSA_CCB = [TPASK_LENGTH0 + 28. 0. 32. 0]. Address of CCB  

* NMLSB_DTYPE = [TPASK_LENGTH0 + 32. 0. 08. 0]. Variable datatype  

* NMLSV_SUBSTRING = [TPASK_LENGTH0 + 33. 0. 01. 0]. Set if substring  

* NMLSV_IMAG = [TPASK_LENGTH0 + 33. 1. 01. 0]. Set if imaginary part

```

R 11  
15-Sep-1984 23:46:38  
15-Sep-1984 22:45:13

VAX-11 Bliss-32 V4.0-742  
\_S255\$DUA28:[FORRTL.SRC]FORNML.REQ;1

Page 17  
(1)

```
: R0560 0
: R0561 0
: R0562 0
: R0563 0
: R0564 0
: R0565 0
: R0566 0
: R0567 0
: R0568 0
: R0569 0
: R0570 0
: R0571 0
: R0572 0
: R0573 0
: R0574 0
: R0575 0
: R0576 0
: R0577 0
:
:     NMLSV_VALUE_IDENT = [TPASK_LENGTHH0 + 33, 2, 01, 0];
:     NMLSV_SUBSCRIPT = [TPASK_LENGTHH0 + 33, 3, 01, 0];
:     NMLSB_CONSTYPE = [TPASK_LENGTHH0 + 34, 0, 08, 0];
:     NMLSL_CURIDX = [TPASK_LENGTHH0 + 36, 0, 32, 0];
:     NMLSL_SUBSCR = [TPASK_LENGTHH0 + 40, 0, 32, 0];
:     NMLSL_SUBSTRLO = [TPASK_LENGTHH0 + 40, 0, 32, 0];
:     NMLSL_SUBSTRHI = [TPASK_LENGTHH0 + 44, 0, 32, 0];
:     NMLSL_CONSBLOCK = [TPASK_LENGTHH0 + 68, 0, 32, 0];
:     NMLSL_REPEATCT = [TPASK_LENGTHH0 + 84, 0, 32, 0];
:     NMLST_TOKEN = [TPASK_LENGTHH0 + 88, 0, 08, 0]
:
:     | Set if last token was an identifier
:     | Set of a subscript has been seen
:     | Constant type
:     | Current index number
:     | Subscripts (7 longwords
:     | Low substring column
:     | High substring column
:     | Constant storage block
:     | 4 longwords
:     | Repeat count
:     | Last token if
:     | it could be an
:     | identifier.
:     | 32 bytes long.
:
:     TES:
:
:     ! End of FORNML.REQ
```

E 11  
12-Sep-1984 23:44:38  
15-Sep-1984 22:44:59

VAX-11 Bliss-32 V4.0-742  
S255SDUA28:[FORRTL.SRC]FORLIB.REQ;1 Page 18 (1)

0578 0  
0579 0    REQUIRE 'RTLIN:FOROPN':

! OPEN definitions

R0580 0  
R0581 0  
R0582 0  
R0583 0  
R0584 0  
R0585 0  
R0586 0  
R0587 0  
R0588 0  
R0589 0  
R0590 0  
R0591 0  
R0592 0  
R0593 0  
R0594 0  
R0595 0  
R0596 0  
R0597 0  
R0598 0  
R0599 0  
R0600 0  
R0601 0  
R0602 0  
R0603 0  
R0604 0  
R0605 0  
R0606 0  
R0607 0  
R0608 0  
R0609 0  
R0610 0  
R0611 0  
R0612 0  
R0613 0  
R0614 0  
R0615 0  
R0616 0  
R0617 0  
R0618 0  
R0619 0  
R0620 0  
R0621 0  
R0622 0  
R0623 0  
R0624 0  
R0625 0  
R0626 0  
R0627 0  
R0628 0  
R0629 0  
R0630 0  
R0631 0  
R0632 0  
R0633 0  
R0634 0  
R0635 0  
R0636 0

♦ This file, FOROPN.REQ, defines the VAX-11 FORTRAN OPEN, CLOSE and INQUIRE keywords and literal values. Edit: SBL1023

\*\*\*\*\*

♦ COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
♦ DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
♦ ALL RIGHTS RESERVED.

♦ THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
♦ ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
♦ INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
♦ COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
♦ OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
♦ TRANSFERRED.

♦ THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
♦ AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
♦ CORPORATION.

♦ DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
♦ SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

\*\*\*\*\*

0-21 - Move parameter encoding symbols for FOR\$S1OBEG to FPAR. TNH 30-May-78  
0-22 = And symbol for statement types. TNH 30-May-78  
0-23 = Change name to FOROPN.REQ JBS 14-NOV-78  
1-001 - Increment version number and add copyright notice JBS 16-NOV-78  
1-002 - Add Some symbolics to define the special LUN numbers for Basic  
PRINT, INPUT, READ. DGP 05-Dec-78  
1-003 - Add necessary symbolics for ISAM. SBL 06-Dec-78  
1-004 - Change file name from FOROPN.REQ to OTSOPN.REQ. JBS 06-DEC-78  
1-005 - Remove the statement type constants. Moved to LUB. DGP 06-Dec-78  
1-006 - Change back to FOROPN. Move more constants to LUB. DGP 08-Dec-78  
1-007 - Fix some comments to reflect the change back to FOROPN. JBS 12-DEC-78  
1-008 - Add DISP='SUBMIT'. SBL 09-Feb-1979  
1-009 - Change ORG, IND to ORG, IDX. Add new arg type. SBL 03-Apr-79  
1-010 - Add new definitions for ISAM. SBL 6-Apr-79  
1-011 - Give BLANK= literal values. SBL 12-Apr-79  
1-012 - Add OPENSK\_DIS SUDE and OPENSK\_DIS PRDE. SBL 19-Apr-79  
1-013 - Add OPENSK\_IOSTAT L. SBL 27-Apr-79  
1-014 - Add INQUIRE keywords. SBL 27-Apr-79  
1-015 - Because OPEN and CLOSE use the IOSTAT keywords, make the  
symbols that control the size of their keyword arrays  
cover them. JBS 01-MAY-1979  
1-016 - Reassign INQUIRE keywords. SBL 01-May-1979  
1-017 - Add INQUIRE keyword values for ORGANIZATION, RECORDTYPE  
and KEYED. SBL 2-Aug-1979  
1-018 - Add OPENSK ARG B R. SBL 7-August-1979  
1-019 - Remove PRINT statement, for new BLISS compiler. JBS 02-OCT-1979  
1-020 - Add CARRIAGECONTROL for INQUIRE. SBL 4-Dec-1979

R0637 0 | 1-021 - ISAM KEY positions in the OPEN arg list are signed longwords  
 R0638 0 | not words. The key length should be an unsigned byte. SBL 12-Mar-1980  
 R0639 0 | 1-022 - Add OPENSK\_DEFAULTF and INOSK\_DEFAULTF. JAW 30-Jun-1981  
 R0640 0 | 1-023 - Add STREAM, STREAM\_CR and STREAM\_LF values for RECORDTYPE. SBL 1-Mar-1983  
 R0641 0 |  
 R0642 0 | --  
 R0643 0 | +  
 R0644 0 | Define symbols for FORTRAN OPEN keywords of form: OPENSK\_symbol  
 R0645 0 | Define literal values of form: OPENSK\_abc\_xyz where abc is  
 R0646 0 | first three letters of keyword and xyz are the first three  
 R0647 0 | letters of the literal.  
 R0648 0 | Define symbols in alphabetical order.  
 R0649 0 | -  
 R0650 0 |  
 R0651 0 | LITERAL  
 R0652 0 | OPENSK\_ACCESS = 4,  
 R0653 0 | OPENSK\_ACC\_DIR = 1,  
 R0654 0 | OPENSK\_ACC\_SEQ = 2,  
 R0655 0 | OPENSK\_ACC\_APP = 3,  
 R0656 0 | OPENSK\_ACC\_KEY = 4,  
 R0657 0 | OPENSK\_ASSOCIAT = T7,  
 R0658 0 | OPENSK\_ASSOC\_L = 0.  
 R0659 0 |  
 R0660 0 |  
 R0661 0 |  
 R0662 0 |  
 R0663 0 | OPENSK\_BLANK = 24,  
 R0664 0 | OPENSK\_BLK\_ZER = 1,  
 R0665 0 | OPENSK\_BLK\_NUL = 2,  
 R0666 0 | OPENSK\_BLOCKSZ = T8,  
 R0667 0 | OPENSK\_BUFFERCO = 9,  
 R0668 0 | OPENSK\_CARRIAGE = 7,  
 R0669 0 | OPENSK\_CAR\_FOR = 1,  
 R0670 0 | OPENSK\_CAR\_LIS = 2,  
 R0671 0 | OPENSK\_CAR\_NON = 3,  
 R0672 0 | OPENSK\_DEFAULTF = 26,  
 R0673 0 | OPENSK\_DISPOSE = 2,  
 R0674 0 | OPENSK\_DIS\_SAV = 1,  
 R0675 0 | OPENSK\_DIS\_DEL = 2,  
 R0676 0 | OPENSK\_DIS\_PRI = 3,  
 R0677 0 | OPENSK\_DIS\_SUB = 4,  
 R0678 0 | OPENSK\_DIS\_PRDE = 5,  
 R0679 0 | OPENSK\_DIS\_SUDE = 6,  
 R0680 0 | OPENSK\_ERR = 3  
 R0681 0 | OPENSK\_EXTENDS = 11,  
 R0682 0 | OPENSK\_FORM = 5,  
 R0683 0 | OPENSK\_FOR\_FOR = 1,  
 R0684 0 | OPENSK\_FOR\_UNF = 2,  
 R0685 0 | OPENSK\_FOR\_UNS = -1,  
 R0686 0 |  
 R0687 0 |  
 R0688 0 | OPENSK\_INITIALIZ = 10,  
 R0689 0 | OPENSK\_IOSTAT = 22,  
 R0690 0 | OPENSK\_IOSTAT\_L = 25,  
 R0691 0 |  
 R0692 0 |  
 R0693 0 | OPENSK\_KEY = 23.

| ACCESS  
 | = 'DIRECT'  
 | = 'SEQUENTIAL'  
 | = 'APPEND'  
 | = 'KEYED'  
 | ASSOCIATEDVARIABLE  
 | 1 if associated variable is a longword  
 | 0 if just a word. Note: this parameter  
 | is not generated by the compiler!  
 | It is needed after all keywords are converted to  
 | 32-bit values.  
 | BLANK  
 | = 'ZERO'  
 | = 'NULL'  
 | BLOCKSIZE  
 | BUFFERCOUNT  
 | CARRIAGE CONTROL  
 | = 'FORTRAN'  
 | = 'LIST'  
 | = 'NONE'  
 | DEFAULTFILE  
 | DISPOSE  
 | = 'SAVE'  
 | = 'DELETE'  
 | = 'PRINT'  
 | = 'SUBMIT'  
 | = 'PRINT/DELETE'  
 | = 'SUBMIT/DELETE'  
 | ERR  
 | EXTENDSIZE  
 | ! FORM  
 | = 'FORMATTED'  
 | = 'UNFORMATTED'  
 | = 'UNSPECIFIED'  
 | Note: this is not generated by compiler.  
 | It is used by default OPEN only.  
 | INITIALSIZE  
 | IOSTAT  
 | 1 If IOSTAT is a longword,  
 | 0 if a word. This is not generated  
 | by the compiler.  
 | KEY

H 11  
15-Sep-1984 23:44:38  
15-Sep-1984 22:45:18

VAX-11 Bliss-32 V4.0-742  
\$255\$DUA28:[FORRTL.SRC]FOROPN.REQ;1

Page 21  
(1)

R0694 0  
R0695 0  
R0696 0  
R0697 0  
R0698 0  
R0699 0  
R0700 0  
R0701 0  
R0702 0  
R0703 0  
R0704 0  
R0705 0  
R0706 0  
R0707 0  
R0708 0  
R0709 0  
R0710 0  
R0711 0  
R0712 0  
R0713 0  
R0714 0  
R0715 0  
R0716 0  
R0717 0  
R0718 0  
R0719 0  
R0720 0  
R0721 0  
R0722 0  
R0723 0  
R0724 0  
R0725 0  
R0726 0  
R0727 0  
R0728 0  
R0729 0  
R0730 0  
R0731 0  
R0732 0  
R0733 0  
R0734 0  
R0735 0  
R0736 0  
R0737 0  
R0738 0  
R0739 0  
R0740 0  
R0741 0  
R0742 0  
R0743 0  
R0744 0  
R0745 0  
R0746 0  
R0747 0  
R0748 0  
R0749 0  
R0750 0

OPENSK\_MAXREC = 16.  
OPENSK\_NAME = 14.  
OPENSK\_NOSPANBL = 12.  
OPENSK\_USEROPEN = 21.  
OPENSK\_ORGANIZA = 19.  
OPENSK\_ORG\_SEQ = 1.  
OPENSK\_ORG\_REL = 2.  
OPENSK\_ORG\_IDX = 3.  
OPENSK\_ORG\_HAS = 4.  
OPENSK\_ORG\_STR = 5.  
OPENSK\_READONLY = 8.  
OPENSK\_RECORDTY = 20.  
OPENSK\_REC\_FIX = 1.  
OPENSK\_REC\_VAR = 2.  
OPENSK\_REC\_SEGM = 3.  
OPENSK\_REC\_STM = 4.  
OPENSK\_REC\_STMCR = 5.  
OPENSK\_REC\_STMLF = 6.  
OPENSK\_RECORDSI = 6.  
OPENSK\_SHARED = 13.  
OPENSK\_TYPE = 15.  
OPENSK\_TYP\_OLD = 1.  
OPENSK\_TYP\_NEW = 2.  
OPENSK\_TYP\_SCR = 3.  
OPENSK\_TYP\_UNK = 4.  
OPENSK\_UNIT = 1.

MAXREC  
NAME  
NOSPANBLOCKS  
USEROPEN  
ORGANIZATION  
= 'SEQUENTIAL'  
= 'RELATIVE'  
= 'INDEXED'  
= 'HASHED'  
= 'STREAM'  
READONLY  
RECORDTYPE  
= 'FIXED'  
= 'VARIABLE'  
= 'SEGMENTED'  
= 'STREAM'  
= 'STREAM-CR'  
= 'STREAM-LF'  
RECORDSIZE  
SHARED  
TYPE  
= 'OLD'  
= 'NEW'  
= 'SCRATCH'  
= 'UNKNOWN'  
UNIT

OPENSK\_KEY\_MAX = OPENSK\_DEFAULTF. ! Max. open parameter  
CLOSSK\_KEY\_MAX = OPENSK\_DEFAULTF; ! Max. CLOSE parameter

Key numbers 27-29 are reserved for future OPEN/CLOSE use.

INQUIRE keyword definitions

LITERAL

INQSK\_FILE = OPENSK\_NAME.  
INQSK\_DEFAULTF = OPENSK\_DEFAULTF.  
INQSK\_UNIT = OPENSK\_UNIT.  
INQSK\_IOSTAT = OPENSK\_IOSTAT.  
INQSK\_IOSTAT\_L = OPENSK\_IOSTAT\_L.  
INQSK\_ERR = OPENSK\_ERR.  
INQSK\_EXIST = 30.  
INQSK\_OPENED = 31.  
INQSK\_NUMBER = 32.  
INQSK\_NAMED = 33.  
INQSK\_NAME = 34.  
INQSK\_ACCESS = 35.  
INQSK\_SEQUENTIA = 36.  
INQSK\_DIRECT = 37.  
INQSK\_FORM = 38.  
INQSK\_FORMATTED = 39.  
INQSK\_UNFORMATT = 40.

Input file name  
Defaultfile  
Input unit number  
IOSTAT  
1 if IOSTAT is a longword  
0 if a word  
1 if ERR= present  
File exists?  
File opened?  
Open on what unit?  
Does it have a name?  
What's its name?  
Access mode?  
Is it sequential?  
Is it direct?  
What's the form?  
Formatted?  
Unformatted?

```

R0751 0
R0752 0
R0753 0
R0754 0
R0755 0
R0756 0
R0757 0
R0758 0
R0759 0
R0760 0
R0761 0
R0762 0
R0763 0
R0764 0
R0765 0
R0766 0
R0767 0
R0768 0
R0769 0
R0770 0
R0771 0
R0772 0
R0773 0
R0774 0
R0775 0
R0776 0
R0777 0
R0778 0
R0779 0
R0780 0
R0781 0
R0782 0
R0783 0
R0784 0
R0785 0
R0786 0
R0787 0
R0788 0
R0789 0
R0790 0
R0791 0
R0792 0
R0793 0
R0794 0
R0795 0
R0796 0
R0797 0
R0798 0
R0799 0
R0800 0
R0801 0
R0802 0
R0803 0
R0804 0
R0805 0
R0806 0
R0807 0

INOSK_RECL = 41,          ! What's the recordsize?
INOSK_NEXTREC = 42,       ! What's the next record
INOSK_BLANK = 43,         ! What are blanks?
INOSK_ORGANIZAT = 44,     ! What's the organization?
INOSK_RECORDTYP = 45,      ! What's the recordtype?
INOSK_KEYED = 46,          ! KEYED allowed?
INOSK_CARRIAGE = 47,       ! What's the carriage control?

INOSK_KEY_MAX = INOSK_CARRIAGE;

!+ Define FORTRAN OPEN argument type codes.
!- Used in field OPENSB_ARG_TYPE

LITERAL
OPENSK_ARG_NULL = 0,       ! keyword with no value
OPENSK_ARG_LIT = 1,         ! literal value in W_INFO WORD
OPENSK_ARG_W_V = 2,         ! expression in W_INFO_WORD
OPENSK_ARG_W_R = 3,         ! next actual is adr. of word
OPENSK_ARG_L_V = 4,         ! next actual is longword value
OPENSK_ARG_L_R = 5,         ! next actual is adr. of longword value
OPENSK_ARG_TZ_R = 6,        ! next actual is adr. of ASCII string
                           ! (needed for compatibility - descriptor
                           ! is the preferred form)
OPENSK_ARG_T_DS = 7,        ! next actual is adr. of string descriptor
OPENSK_ARG_ZI = 8,          ! next actual is proc. adr.
OPENSK_ARG_INLN = 9,         ! next INFO_WORD longwords are arg.
OPENSK_ARG_B_R = 10,        ! next actual is address of byte

OPENSK_ARG_MAX = OPENSK_ARG_B_R; ! max. arg type code

!+ Define fields within FORTRAN OPEN parameters
!- MACRO

OPENSB_KEY      = 0,0,8,0 %, ! keyword code. Codes are of form:
OPENSB_ARG_TYPE = 0,8,8,0 %, ! OPENSK_keyname
OPENSB_INFO     = 0,16,16,1 %, ! arg type code. Codes are
                               ! of form: OPENSK_ARG_type
OPENSA_VALUE    = 0,0,%BPADDR,0 %, ! 16-bit information.
OPENSG_VALUE    = 0,0,%BPVAL,0 %, ! sign extend to 32-bits.
                               ! Address of value - in next
                               ! position in parameter list
                               ! General value - in next
                               ! position in parameter list

!+ Macros and literals for KEY= keyword and for ISAM
!- LITERAL
OPENSK_XAB_SIZE = XABSC_KEYLEN + 4;
MACRO

```

15-Sep-1984 23:44:38  
15-Sep-1984 22:45:18

VAX-11 Bliss-32 V4.0-742  
\$255\$DUA28:[FORRTL.SRC]FOROPN.REQ;1

Page 23  
(1)

R0808 0 OPENSB\_DTYPE = 0,0,8,0%; ! Key datatype in OPEN list  
R0809 0 OPENSL\_KEY\_LO = 4,0,32,1%; ! Low key position in OPEN list  
R0810 0 OPENSL\_KEY\_HI = 8,0,32,1%; ! High key position in OPEN list  
R0811 0 OPENSB\_KTYPE = XABSC\_KEYLEN,0,8,0%; ! Saved datatype  
R0812 0 OPENSB\_SIZE = XABSC\_KEYLEN,8,8,0%; ! Saved key size  
R0813 0 OPENSW\_POS0 = XABSC\_KEYLEN,16,16,0%; ! Saved low position  
R0814 0  
R0815 0  
R0816 0 !+ Max. length of ASCII string for FORTRAN OPEN file name array  
R0817 0 !-  
R0818 0  
R0819 0 LITERAL  
R0820 0 OPENSK\_STR\_MAX = 100; ! Max. length of an ASCII string  
R0821 0 (arg type TZ\_R only). No limit  
R0822 0 for string descriptor strings  
R0823 0 !+  
R0824 0 Constants used in parameter encoding between the I/O statement routines  
R0825 0 and routine FORSSID\_BEG. The codes are  
R0826 0 both bit positions in the flag word and an index into  
R0827 0 a table used for sorting out the parameters.  
R0828 0 All are optional for some I/O statement.  
R0829 0 !-  
R0830 0  
R0831 0 LITERAL  
R0832 0 K\_UNIT = 0. ! user supplied unit number  
R0833 0 K\_CHAR\_COUNT = 1. ! size of user supplied record for EN/DECODE  
R0834 0 K\_REC\_NO = 2. ! user supplied record number  
R0835 0 K\_FMT\_ADR = 3. ! user supplied format address  
R0836 0 K\_USR\_BUF\_ADR = 4. ! user supplied buffer for EN/DECODE  
R0837 0 K\_OBJ\_TIME\_FMT = 7; ! bit says object time format  
R0838 0  
R0839 0  
R0840 0  
R0841 0 ! End of file FOROPN.REQ

K 11  
15-Sep-1984 23:44:38  
15-Sep-1984 22:44:59

VAX-11 Bliss-32 V4.0-742  
\$255\$DUA28:[FORRTL.SRC]FORLIB.REQ;1

Page 24  
(1)

0842 0  
0843 0

REQUIRE 'RTLML:FORPAR';

! Assorted definitions

L 11  
15-Sep-1984 23:44:38  
15-Sep-1984 22:46:18

VAX-11 Bliss-32 V4.0-742  
\$255\$DUA28:[FORRTL.OBJ]FORPAR.R32:1

Page 25  
(1)

R0844 0 | \*\*\*\*\*  
R0845 0 | \*\*\*\*\*  
R0846 0 | \*\*\*\*\*  
R0847 0 | \*\*\*\*\*  
R0848 0 | \*\*\*\*\*  
R0849 0 | \*\*\* MODULE \$FORPAR \*\*\*  
R0850 0 | literal FORSK\_CLASS\_SB = 191;  
R0851 0 | literal FORSK\_CLASS\_NL = 190;  
R0852 0 | literal FORSK\_CONTROL\_Z = 26;  
R0853 0 | literal FORSK\_UNWINDPOP = 0;  
R0854 0 | literal FORSK\_UNWINDNOP = 1;  
R0855 0 | literal FORSK\_UNWINDRET = 2;  
R0856 0 | literal FORSS\_FORSR\_PAR = 2;  
R0857 0 | macro FORSr\_union\_1 = 0,0,16,0 %;  
R0858 0 | literal FORSS\_union\_1 = 2;  
R0859 0 | macro FORSr\_structure\_1 = 0,0,16,0 %;  
R0860 0 | literal FORSS\_structure\_1 = 2;  
R0861 0 | macro FORSB\_STMT\_TYPE = 0,0,8,0 %;  
R0862 0 | macro FORSB\_STMT\_FLAGS = 1,0,8,0 %;  
R0863 0 | macro FORSr\_structure\_2 = 0,0,16,0 %;  
R0864 0 | literal FORSS\_structure\_2 = 2;  
R0865 0 | macro FORSV\_OBJ\_FMT = 0,8,1,0 %;

M 11  
15-Sep-1984 23:44:38  
15-Sep-1984 22:44:59

VAX-11 Bliss-32 V4.0-742  
S255\$DUA28:[FORRTL.SRC]FORLIB.REQ;1 Page 26 (1)

0866 0  
0867 0 REQUIRE 'RTLML:FORRCE';

! RFA Cache Entry structure

```

R0868 0
R0869 0
R0870 0
R0871 0
R0872 0
R0873 0
R0874 0
R0875 0
R0876 0
R0877 0
R0878 0
R0879 0
R0880 0
R0881 0
R0882 0
R0883 0
R0884 0
R0885 0
R0886 0
R0887 0
R0888 0
R0889 0
R0890 0
R0891 0
R0892 0
R0893 0
R0894 0
R0895 0
R0896 0
R0897 0

***** (Created 15-SEP-1984 22:46:24 by VAX-11 SDL V2.0) ***** Source: 15-SEP-1984 22:45:34 $255$DUA28:[FORRTL.SRC]FORR
***** MODULE RCEDEF IDENT 1-001 ***
+
An RFA Cache Entry (RCE) contains information about previous records
in the file for use by FORSBACKSPACE, which implements the FORTRAN
BACKSPACE statement.
For sequential organization and access disk files, each time a new record
is read or written, an entry is added to the RFA cache. The cache itself
is a circularly-linked list, established when the file is opened.
literal RCE_K_CACHE_SIZE = 20;           ! Number of entries in cache
literal RCE_S_RFA_UNION = 8;
FIELD RCE_STRUCT$FIELDSET =
SET
RCE_A_NEXT = [0,0,32,0] .           ! Pointer to next entry
RCE_A_PREV = [4,0,32,0] .           ! Pointer to previous entry
RCE_L_LOG_RECNO = [8,0,32,0] .     ! Logical record number for this entry
RCE_Q_RFA = [12,0,0,0] .           ! RFA for this entry
RCE_L_RFA0 = [12,0,32,0] .         ! First 4 bytes of RFA
RCE_W_RFA4 = [16,0,16,0] .         ! Last 2 bytes of RFA
RCE_R_RFA_STRUCT = [12,0,0,0] .
RCE_R_RFA_UNION = [12,0,0,0] .
TES;
literal RCE_S_RCE_STRUCT = 20;
MACRO RCE_R_RCE_STRUCT = BLOCK [RCE_S_RCE_STRUCT,byte] FIELD (RCE_STRUCT$FIELDSET) %;

```

B 12  
12-Sep-1984 23:44:38  
15-Sep-1984 22:44:59

VAX-11 Bliss-32 V4.0-742  
S255\$DUA28:[FORRTL.SRC]FORLIB.REQ;1 Page 28 (1)

0898 0  
0899 0 REQUIRE 'RTLML:OTSIDB.BLF';

: ! Intra-statement block definitions

```

R0900 0
R0901 0
R0902 0
R0903 0
R0904 0
R0905 0
R0906 0
R0907 0
R0908 0
R0909 0
R0910 0
R0911 0
R0912 0
R0913 0
R0914 0
R0915 0
R0916 0
R0917 0
R0918 0
R0919 0
R0920 0
R0921 0
R0922 0
R0923 0
R0924 0
R0925 0
R0926 0
R0927 0
R0928 0
R0929 0
R0930 0
R0931 0
R0932 0
R0933 0
R0934 0
R0935 0
R0936 0
R0937 0
R0938 0
R0939 0
R0940 0
R0941 0
R0942 0
R0943 0
R0944 0
R0945 0
R0946 0
R0947 0
R0948 0
R0949 0
R0950 0
R0951 0
R0952 0
R0953 0
R0954 0
R0955 0
R0956 0

***** MODULE $ISBDEF ****
literal ISBSK-ST-TY-WSF = 1;
literal ISBSK-FORSTTYLO = 1;
literal ISBSK-ST-TY-RSF = 2;
literal ISBSK-ST-TY-WSU = 3;
literal ISBSK-ST-TY-RSU = 4;
literal ISBSK-ST-TY-WDF = 5;
literal ISBSK-ST-TY-RDF = 6;
literal ISBSK-ST-TY-WDU = 7;
literal ISBSK-ST-TY-RDU = 8;
literal ISBSK-ST-TY-WSL = 9;
literal ISBSK-ST-TY-RSL = 10;
literal ISBSK-ST-TY-WMF = 11;
literal ISBSK-MIN DE EN = 11;
literal ISBSK-ST-TY-RMF = 12;
literal ISBSK-MAX DE EN = 12;
literal ISBSK-ST-TY-QXF = 13;
literal ISBSK-ST-TY-RKF = 14;
literal ISBSK-ST-TY-WXU = 15;
literal ISBSK-ST-TY-RKU = 16;
literal ISBSK-ST-TY-WIF = 17;
literal ISBSK-ST-TY-RIF = 18;
literal ISBSK-ST-TY-WSN = 19;
literal ISBSK-ST-TY-RSN = 20;
literal ISBSK-ST-TY-WIL = 21;
literal ISBSK-ST-TY-RIL = 22;
literal ISBSK-FORSTTYHI = 22;
literal ISBSK-ST-TY-PRI = 27;
literal ISBSK-BASSTTYLO = 27;
literal ISBSK-ST-TY-LIN = 28;
literal ISBSK-ST-TY-PSE = 29;
literal ISBSK-ST-TY-INP = 30;
literal ISBSK-ST-TY-PRU = 31;
literal ISBSK-ST-TY-INAL = 32;
literal ISBSK-ST-TY-DEL = 33;
literal ISBSK-ST-TY-REA = 34;
literal ISBSK-ST-TY-UPD = 35;
literal ISBSK-ST-TY-GSE = 36;
literal ISBSK-ST-TY-RES = 37;
literal ISBSK-ST-TY-SCR = 38;
literal ISBSK-ST-TY-PRE = 39;
literal ISBSK-ST-TY-GRE = 40;
literal ISBSK-ST-TY-FRE = 41;
literal ISBSK-ST-TY-UNL = 42;
literal ISBSK-ST-TY-FEE = 43;
literal ISBSK-ST-TY-GIN = 44;
literal ISBSK-ST-TY-PIN = 45;
literal ISBSK-ST-TY-MOV = 46;
literal ISBSK-ST-TY-FIN = 47;
literal ISBSK-ST-TY-MIN = 48;
literal ISBSK-ST-TY-RIN = 49;
literal ISBSK-ST-TY-MLI = 50;

```

0 12  
15-Sep-1984 23:44:38  
15-Sep-1984 23:02:23

VAX-11 Bliss-32 V4.0-742  
\$255SDUA28:[FORRTL.OBJ]OTSISB.BLF;1

Page 30  
(1)

```
R0957 0 literal ISBSK-ST-TY-FSE = 51;  
R0958 0 literal ISBSK-ST-TY-MPR = 53;  
R0959 0 literal ISBSK-ST-TY-MRE = 54;  
R0960 0 literal ISBSK-ST-TY-GRFA = 55;  
R0961 0 literal ISBSK-ST-TY-FRFA = 56;  
R0962 0 literal ISBSK-BASSTTYHI = 56;  
R0963 0 literal ISBSK-NEG_LUB = -100;  
R0964 0 literal ISBSK-ISB-LEN = 188;  
R0965 0 literal ISBSS-union_1B = 16;  
R0966 0 literal ISBSS-union_2 = 16;  
R0967 0 literal ISBSS-lub_filler = 100;  
R0968 0 FIELD ISBSFIE[DSET =  
R0969 0 SET  
R0970 0 ISBSA_RESTARTPC = [-188,0,32,0] ;  
R0971 0 ISBSA_USR_HANDL = [-188,0,32,0] ;  
R0972 0 ISBSr_union_1 = [-188,0,32,0] ;  
R0973 0 ISBSA_MAJ_F_PTR = [-184,0,32,0] ;  
R0974 0 ISBSA_PREVIOUS_LUB = [-184,0,32,0] ;  
R0975 0 ISBSr_union_1A = [-184,0,32,0] ;  
R0976 0 ISBSA_USER_PP = [-180,0,32,0] ;  
R0977 0 ISBSW_FMT_STKP = [-176,0,0,0] ;  
R0978 0 ISBSA_SAVE_PTR = [-176,0,32,0] ;  
R0979 0 ISBSA_SAVE-END = [-172,0,32,0] ;  
R0980 0 !ISBSr_structre_1B = [-176,0,0,0] ;  
R0981 0 ISBSr_union_1B = [-176,0,0,0] ;  
R0982 0 ISBSW_FMT_STKR = [-160,0,0,0] ;  
R0983 0 ISBSB-SCA-FAC_D = [-160,0,0,0] ;  
R0984 0 ISBSr_union_2 = [-160,0,0,0] ;  
R0985 0 ISBSB_ERR_NO = [-144,0,8,0] ;  
R0986 0 ISBSB-SCA[E_FAC = [-144,0,8,1] ;  
R0987 0 ISBSr_union_3 = [-144,0,8,0] ;  
R0988 0 ISBSB_STTM_TYPE = [-143,0,8,0] ;  
R0989 0 ISBSW_FMT[EN = [-142,0,16,0] ;  
R0990 0 ISBSA_ERR-EQUAL = [-140,0,32,0] ;  
R0991 0 ISBSA-END-EQUAL = [-136,0,32,0] ;  
R0992 0 ISBSA_FMT-BEG = [-132,0,32,0] ;  
R0993 0 ISBSA_FMT_PTR = [-128,0,32,0] ;  
R0994 0 ISBSL_LIS-HEAP_LEN = [-128,0,32,0] ;  
R0995 0 ISBSr_union_3A = [-128,0,32,0] ;  
R0996 0 ISBSA_LIS_STR = [-124,0,32,0] ;  
R0997 0 ISBSB_FMT-P = [-120,0,8,1] ;  
R0998 0 ISBSW_FMT-U = [-119,0,16,0] ;  
R0999 0 ISBSB_FMT-D = [-117,0,8,0] ;  
R1000 0 ISBSB_FMT-E = [-116,0,8,0] ;  
R1001 0 ISBSW_FMT-REP = [-115,0,16,1] ;  
R1002 0 ISBSW_LIS-REP = [-115,0,16,0] ;  
R1003 0 ISBSW_LEN-REM = [-115,0,16,0] ;  
R1004 0 ISBSr_union_4 = [-115,0,16,0] ;  
R1005 0 ISBSB_FMT_CODE = [-113,0,8,0] ;  
R1006 0 ISBSV_FMT_REPRE = [-113,0,8,0] ;  
R1007 0 ISBSR_FMT_CODE_STRUCT = [-113,0,8,0] ;  
R1008 0 ISBSR_FMT_CODE_UNION = [-113,0,8,0] ;  
R1009 0 ISBSB_LIS-CTYPE = [-113,0,8,0] ;  
R1010 0 ISBSr_union_5 = [-113,0,8,0] ;  
R1011 0 ISBSW_FMT_REVER = [-112,0,16,0] ;  
R1012 0 ISBSB_FMT-DEP = [-110,0,8,0] ;  
R1013 0 ISBSW_FMT_FLAGS = [-109,0,16,0] ;
```

R1014 0 ISB\$B INP FLAGS = [-109,0,8,0] .  
R1015 0 ISB\$V BN = [-109,0,1,0] .  
R1016 0 ISB\$V ONLY E = [-109,1,1,0] .  
R1017 0 ISB\$V ERR OFLO = [-109,2,1,0] .  
R1018 0 ISB\$V DONTROUND = [-109,3,1,0] .  
R1019 0 ISB\$V SKIPTABS = [-109,4,1,0] .  
R1020 0 ISB\$V EXP LETTER = [-109,5,1,0] .  
R1021 0 ISB\$V FORCESCALE = [-109,6,1,0] .  
R1022 0 ISBSR INP FLAGS STRUCT = [-109,0,8,0] .  
R1023 0 ISBSR INP FLAGS UNION = [-109,0,8,0] .  
R1024 0 ISB\$B OUT FLAGS = [-108,0,8,0] .  
R1025 0 ISB\$V SP = [-108,0,1,0] .  
R1026 0 ISB\$V ERR OFLO = [-108,1,1,0] .  
R1027 0 ISBSR OUT FLAGS STRUCT = [-108,0,8,0] .  
R1028 0 ISBSR OUT FLAGS UNION = [-108,0,8,0] .  
R1029 0 ISBSR FMT FLAGS STRUCT = [-109,0,16,0] .  
R1030 0 ISBSR FMT FLAGS UNION = [-109,0,16,0] .  
R1031 0 ISBSR STTM STAT = [-106,0,16,0] .  
R1032 0 ISB\$V\_P FORM\_CH = [-106,0,2,0] .  
R1033 0 ISB\$V\_DOLLAR = [-106,2,1,0] .  
R1034 0 ISB\$V\_USER ELEM = [-106,3,1,0] .  
R1035 0 ISB\$V\_SLASH = [-106,4,1,0] .  
R1036 0 ISB\$V\_LAST REC = [-106,5,1,0] .  
R1037 0 ISB\$V\_DE ENCODE = [-106,6,1,0] .  
R1038 0 ISB\$V\_L15 HEAP = [-106,7,1,0] .  
R1039 0 ISB\$V\_RECURSIVE = [-106,8,1,0] .  
R1040 0 ISB\$V\_MAT CONT = [-106,9,1,0] .  
R1041 0 ISB\$V\_MAT PRINT = [-106,10,1,0] .  
R1042 0 ISB\$V\_PRINTINI = [-106,11,1,0] .  
R1043 0 ISB\$V\_SNGL\_ELEM = [-106,12,1,0] .  
R1044 0 ISB\$V\_NEED INIT = [-106,13,1,0] .  
R1045 0 ISBSR STTM STAT STRUCTURE = [-106,0,16,0] .  
R1046 0 ISBSR STTM STAT UNION = [-106,0,16,0] .  
R1047 0 ISBSA\_INTFILEEND = [-104,0,32,0]  
R1048 0 TES;  
R1049 0 literal ISB\$S\_ISB = 189;  
R1050 0 MACRO ISB = B[OCK [ISB\$S\_ISB,byte] FIELD (ISB\$FIELDSET) X;

F 12  
15-Sep-1984 23:44:38  
15-Sep-1984 22:44:59

VAX-11 Bliss-32 V4.0-742  
S255\$DUA28:[FORRTL.SRC]FORLIB.REQ;1 Page 32 (1)

1051 0 UNDECLARE %QUOTE ISB;  
1052 0  
1053 0 REQUIRE 'RTLML:OTSLUB.BLF';

! Logical Unit Block definitions

R1054 0  
R1055 0  
R1056 0  
R1057 0  
R1058 0  
R1059 0  
R1060 0  
R1061 0  
R1062 0  
R1063 0  
R1064 0  
R1065 0  
R1066 0  
R1067 0  
R1068 0  
R1069 0  
R1070 0  
R1071 0  
R1072 0  
R1073 0  
R1074 0  
R1075 0  
R1076 0  
R1077 0  
R1078 0  
R1079 0  
R1080 0  
R1081 0  
R1082 0  
R1083 0  
R1084 0  
R1085 0  
R1086 0  
R1087 0  
R1088 0  
R1089 0  
R1090 0  
R1091 0  
R1092 0  
R1093 0  
R1094 0  
R1095 0  
R1096 0  
R1097 0  
R1098 0  
R1099 0  
R1100 0  
R1101 0  
R1102 0  
R1103 0  
R1104 0  
R1105 0  
R1106 0  
R1107 0  
R1108 0  
R1109 0  
R1110 0  
\*\*\*\*\*  
Created 15-SEP-1984 22:49:15 by VAX-11 SDL V2.0 Source: 15-SEP-1984 22:47:02 \$255\$DUA28:[LIBRTL.SRC]OTSL  
\*\*\*\*\*  
\*\*\* MODULE SLUBDEF \*\*\*  
literal LUBSK\_ORG\_SEQUE = 1;  
literal LUBSK\_ORG\_RELAT = 2;  
literal LUBSK\_ORG\_INDEX = 3;  
literal LUBSK\_ORG\_TERMI = 4;  
literal LUBSK\_ORG\_VIRTU = 5;  
literal LUBSK\_LUN\_BPRI = -8;  
literal LUBSK\_LUN\_INPU = -7;  
literal LUBSK\_LUN\_BREAD = -6;  
literal LUBSK\_LUN\_ENCD = -5;  
literal LUBSK\_LUN\_READ = -4;  
literal LUBSK\_LUN\_ACCE = -3;  
literal LUBSK\_LUN\_TYPE = -2;  
literal LUBSK\_LUN\_PRIN = -1;  
literal LUBSK\_DLUN\_MIN = -8;  
literal LUBSK\_DLUN\_MAX = -1;  
literal LUBSK\_LUN\_MIN = 0;  
literal LUBSK\_LUN\_MAX = 119;  
literal LUBSK\_D\_MARGIN = 72;  
literal LUBSK\_PBUF\_SIZ = 80;  
literal LUBSK\_NEG\_BLN = -100;  
literal LUBSK\_LANG\_MIN = 0;  
literal LUBSK\_LANG\_NONE = 0;  
literal LUBSK\_LANG\_BAS = 1;  
literal LUBSK\_LANG\_FOR = 2;  
literal LUBSK\_LANG\_MAX = 2;  
literal LUBSK\_LUB [EN = 100;  
literal LUB\$S\_QUEUE = 8;  
literal LUB\$S\_DID = 6;  
FIELD LUBSFIE[DSET =  
SET  
LUBSA\_UBF = [-100,0,32,0]  
LUBSW\_UNIT\_STT3 = [-96,0,16,0] .  
LUBSV\_NOECHO = [-96,0,1,0] .  
LUBSV\_ONECHR = [-96,1,1,0] .  
LUBSV\_CCO = [-96,2,1,0] .  
LUBSV\_FIND\_LAST = [-96,3,1,0] .  
LUBSV\_PTA = [-96,4,1,0] .  
LUBSV\_AST\_GUARD = [-96,5,1,0] .  
LUBSV\_CR = [-96,6,1,0] .  
LUBSV\_FTN = [-96,7,1,0] .  
LUBSV\_PRN = [-96,8,1,0] .  
LUBSV\_NOMARGIN = [-96,9,1,0] .  
LUB\$V\_USEROPEN = [-96,10,1,0] .  
LUBSV\_NOTSEQORG = [-96,11,1,0] .  
LUBSV\_ANSI = [-96,12,1,0] .  
LUBSV\_RFA\_CACHE\_ENABLE = [-96,13,1,0] .  
LUBSV\_FIE[D USE = [-96,14,1,0]  
LUBSR\_UNIT\_STT3\_STRUCT = [-96,0,16,0] .  
LUB\$r\_union\_T = [-96,0,16,0] .  
LUBSW\_BLS = [-96,0,16,0] .

H 12  
15-Sep-1984 23:44:38  
15-Sep-1984 23:02:25

VAX-11 Bliss-32 V4.0-742  
\$\_255\$DUA28:[FORRTL.OBJ]OTSLUB.BLF;1 Page 34 (1)

R1111 0 LUBSA CLOSE = [-92.0.32.0] .  
R1112 0 LUBSQ\_QUEUE = [-88.0.0.0] .  
R1113 0 LUBSA\_BUF\_PTR = [-80.0.32.0] .  
R1114 0 LUBSA\_BUF\_END = [-76.0.32.0] .  
R1115 0 LUBSA\_BUDDY\_PTR = [-72.0.32.0] .  
R1116 0 LUBSA\_BUF\_BEG = [-68.0.32.0] .  
R1117 0 LUBSA\_BUF\_HIGH = [-64.0.32.0] .  
R1118 0 LUBSB\_ORGAN = [-60.0.8.0] .  
R1119 0 LUBSB\_BKS = [-59.0.8.0] .  
R1120 0 LUBSW\_LUN = [-58.0.16.1] .  
R1121 0 LUBSL\_PRINT\_POS = [-56.0.32.0] .  
R1122 0 LUBSA\_RFA\_CACHE\_BEG = [-56.0.32.0] .  
R1123 0 LUB\$r\_union\_1A = [-56.0.32.0] .  
R1124 0 LUBSL\_WAIT\_TIME = [-52.0.32.0] .  
R1125 0 LUBSA\_RFA\_CACHE\_PTR = [-52.0.32.0] .  
R1126 0 LUB\$r\_union\_1B = [-52.0.32.0] .  
R1127 0 LUBSWIFI = [-48.0.16.0] .  
R1128 0 LUBSW\_RBUF\_SIZE = [-46.0.16.0] .  
R1129 0 LUBSW\_R\_MARGIN = [-44.0.16.0] .  
R1130 0 LUBSW\_D\_MARGIN = [-42.0.16.0] .  
R1131 0 LUBSB\_LANGUAGE = [-40.0.8.0] .  
R1132 0 LUBSB\_RFIM = [-39.0.8.0] .  
R1133 0 LUBSW\_BAS\_VFC = [-38.0.16.0] .  
R1134 0 LUBSB\_BAS\_VFC1 = [-38.0.8.0] .  
R1135 0 LUBSB\_BAS\_VFC2 = [-37.0.8.0] .  
R1136 0 LUB\$R\_BAS\_VFC\_STRUCT = [-38.0.16.0] .  
R1137 0 LUB\$r\_union\_2 = [-38.0.16.0] .  
R1138 0 LUBSA\_ASSOC\_VAR = [-36.0.32.0] .  
R1139 0 LUBSL\_ALQ = [-36.0.32.0] .  
R1140 0 LUB\$r\_union\_3 = [-36.0.32.0] .  
R1141 0 LUBSL\_LOG\_RECNO = [-32.0.32.0] .  
R1142 0 LUBSL\_REC\_MAX = [-28.0.32.0] .  
R1143 0 LUBSA\_FAB = [-24.0.32.0] .  
R1144 0 LUBSA\_RBUF\_ADDR = [-20.0.32.0] .  
R1145 0 LUBSW\_DID = [-16.0.0.0] .  
R1146 0 LUBSB\_RAT = [-10.0.8.0] .  
R1147 0 LUBSB\_RSL = [-9.0.8.0] .  
R1148 0 LUBSA\_RSN = [-8.0.32.0] .  
R1149 0 LUBSW\_UNIT\_ATTR = [-4.0.16.0] .  
R1150 0 LUBSV\_OPENED = [-4.0.1.0] .  
R1151 0 LUBSV\_IO\_ACTIVE = [-4.1.1.0] .  
R1152 0 LUBSV\_READ\_ONLY = [-4.2.1.0] .  
R1153 0 LUBSV\_OLD\_FILE = [-4.3.1.0] .  
R1154 0 LUBSV\_DIRECT = [-4.4.1.0] .  
R1155 0 LUBSV\_SCRATCH = [-4.5.1.0] .  
R1156 0 LUBSV\_DELETE = [-4.6.1.0] .  
R1157 0 LUBSV\_PRINT = [-4.7.1.0] .  
R1158 0 LUBSV\_FORMATTED = [-4.8.1.0] .  
R1159 0 LUBSV\_UNFORMAT = [-4.9.1.0] .  
R1160 0 LUBSV\_FIXED = [-4.10.1.0] .  
R1161 0 LUBSV\_SEGMENTED = [-4.11.1.0] .  
R1162 0 LUBSV\_ASS\_VAR\_L = [-4.12.1.0] .  
R1163 0 LUBSV\_APPEND = [-4.13.1.0] .  
R1164 0 LUBSV\_SEQUENTIA = [-4.14.1.0] .  
R1165 0 LUBSV\_KEYED = [-4.15.1.0] .  
R1166 0 LUB\$R\_UNIT\_ATTR\_STRUCT = [-4.0.16.0] .  
R1167 0 LUB\$r\_union\_4 = [-4.0.16.0] .

112  
15-Sep-1984 23:44:38  
15-Sep-1984 23:02:25

VAX-11 Bliss-32 v4.0-742  
\$255\$DUA28:[FORRTL.OBJ]OTSLUB.BLF;1 Page 35 (1)

```
R1168 0 LUB$W UNIT STT2 = [-2,0,16,0] ;
R1169 0 LUB$V_VIRT RSN = [-2,0,1,0] ;
R1170 0 LUB$V-ENDFILEOPN = [-2,1,1,0] ;
R1171 0 LUB$V-FORM CHAR = [-2,2,1,0] ;
R1172 0 LUB$V-OUTBUF DR = [-2,3,1,0] ;
R1173 0 LUB$V-TERM FOR = [-2,4,1,0] ;
R1174 0 LUB$V-TERM-DEV = [-2,5,1,0] ;
R1175 0 LUB$V-FORCIBLE = [-2,6,1,0] ;
R1176 0 LUB$V-UNIT 0 = [-2,7,1,0] ;
R1177 0 LUB$V-VA USE = [-2,8,1,0] ;
R1178 0 LUB$V-BLK USE = [-2,9,1,0] ;
R1179 0 LUB$V-M-STREAM = [-2,10,1,0] ;
R1180 0 LUB$V-M-STR C = [-2,11,1,0] ;
R1181 0 LUB$V-DEALLOC = [-2,12,1,0] ;
R1182 0 LUB$V-SUBMIT = [-2,13,1,0] ;
R1183 0 LUB$V-NULLBLNK = [-2,14,1,0] ;
R1184 0 LUB$V-USER RBUF = [-2,15,1,0] ;
R1185 0 LUB$R UNIT STT2 STRUCT = [-2,0,16,0] ;
R1186 0 LUB$R_union_5 = [-2,0,16,0]
R1187 0 TES;
R1188 0 literal LUB$S_LUB = 101;
R1189 0 MACRO LUB = B[OCK [LUB$S_LUB,byte] FIELD (LUB$FIELDSET) %;
```

12  
15-Sep-1984 23:44:38  
15-Sep-1984 22:44:59

VAX-11 Bliss-32 V4.0-742  
\_S255\$DUA28:[FORRTL.SRC]FORLIB.REQ;1 Page 36 (1)

```
1190 0 UNDECLARE %QUOTE LUB;
1191 0
1192 0
1193 0 * Define macro that declares the CCB structure. Note that it has no
1194 0 allocation size - it must be used in a REF declaration.
1195 0
1196 0
1197 0 MACRO
1198 0     $FORSCCB_DECL = BLOCK [, BYTE] FIELD (LUB$FIELDSET, ISBSFIELDSET) %;
1199 0
1200 0 REQUIRE 'RTLIN:OTSCCBREQ';           ! OTS CCB data structure definitions
```

R1201 0  
R1202 0  
R1203 0  
R1204 0  
R1205 0  
R1206 0  
R1207 0  
R1208 0  
R1209 0  
R1210 0  
R1211 0  
R1212 0  
R1213 0  
R1214 0  
R1215 0  
R1216 0  
R1217 0  
R1218 0  
R1219 0  
R1220 0  
R1221 0  
R1222 0  
R1223 0  
R1224 0  
R1225 0  
R1226 0  
R1227 0  
R1228 0  
R1229 0  
R1230 0  
R1231 0  
R1232 0  
R1233 0  
R1234 0  
R1235 0  
R1236 0  
R1237 0  
R1238 0  
R1239 0  
R1240 0  
R1241 0  
R1242 0  
R1243 0  
R1244 0  
R1245 0  
R1246 0  
R1247 0  
R1248 0  
R1249 0  
R1250 0  
R1251 0  
R1252 0  
R1253 0  
R1254 0  
R1255 0  
R1256 0  
R1257 0

+ This file, OTSCCBREQ.REQ, defines the interface to OTSSPUSH\_CCB Edit: SBL1006  
-  
\*\*\*\*\*  
\* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
\* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
\* ALL RIGHTS RESERVED.  
\*  
\* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
\* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
\* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
\* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
\* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
\* TRANSFERRED.  
\*  
\* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
\* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
\* CORPORATION.  
\*  
\* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
\* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  
\*  
\*\*\*\*\*  
  
Revision History:  
1-001 - Original. JBS 09-JAN-1979  
1-002 - Change name to OTSCCBREQ.REQ so as not to conflict at system  
build time with OTSCCB.B32. SBL 10-May-1979  
1-003 - Add the definition of the structure for OTSSSAA\_LUB\_TAB.  
JBS 28-JUN-1979  
1-004 - Add the definition of the field for OTSSSV\_LUN\_OWN.  
JBS 16-AUG-1979  
1-005 - Remove PRINT statement, for the new BLISS compiler.  
JBS 02-OCT-1979  
1-006 - Add new structure OTSSLUN\_OWN\_ST used for OTSSSV\_LUN\_OWN. This  
helps BLISS generate smaller code for references to this structure.  
--  
  
+ Define the return codes from OTSSPUSH\_CCB.  
-  
LITERAL  
OTSSK\_PUSH\_MIN = 1, ! Smallest valid value  
OTSSK\_PUSH\_OK = 1, ! CCB loaded, I/O not active  
OTSSK\_PUSH\_ACT = 2, ! CCB loaded, I/O active on this LUN  
OTSSK\_PUSH\_FAIL = 3, ! CCB not loaded, out of virtual storage  
OTSSK\_PUSH\_MAX = 3; ! Largest valid value  
  
+ The following structure is used for addressing OTSSSAA\_LUB\_TAB.

R1258 0  
R1259 0  
R1260 0  
R1261 0  
R1262 0  
R1263 0  
R1264 0  
R1265 0  
R1266 0  
R1267 0  
R1268 0  
R1269 0  
R1270 0  
R1271 0  
R1272 0  
R1273 0  
R1274 0  
R1275 0  
R1276 0  
R1277 0  
R1278 0  
R1279 0  
R1280 0  
R1281 0  
R1282 0  
R1283 0  
R1284 0  
R1285 0  
R1286 0  
R1287 0  
R1288 0  
R1289 0  
R1290 0  
R1291 0  
R1292 0  
R1293 0  
R1294 0  
R1295 0  
R1296 0  
R1297 0

| It is similar to VECTOR, but offsets the index so that certain  
| negative logical unit numbers can be used, and each element is a  
| quadword so as to act as a queue header.  
|  
STRUCTURE  
  OTSSSLUB\_TAB\_ST [I, SIDE; N, LB, UNIT = 4, EXT = 0] =  
  [N\*UNIT\*2]  
  (OTSSSLUB\_TAB\_ST + ((SIDE + ((I - LB)\*2)\*UNIT))<0, 8\*UNIT, EXT>;  
|  
| The following structure is similar to BLOCKVECTOR, but allows a low and high  
| bound.  
|  
STRUCTURE  
  OTSSSLUN\_OWNR\_ST [I, O, P, S, E; L, H, BS, UNIT=1] =  
  [((H - L) + 1) \* BS \* UNIT]  
  (OTSSSLUN\_OWNR\_ST + (O - L) + ((O + I) \* BS \* UNIT))<P, S, E>;  
|  
| The following field is used to refer to OTSSSV\_LUN\_OWNR, which has a bit  
| for each LUB, each block containing a bit for each language.  
|  
FIELD  
  OTSSSV\_OWNR\_FLD =  
  SET  
  OTSSSV\_OWNR\_BAS = [0, LUBSK\_LANG\_BAS, 1, 0], ! BASIC  
  OTSSSV\_OWNR\_FOR = [0, LUBSK\_LANG\_FOR, 1, 0], ! FORTRAN  
  OTSSSV\_OWNR = [0, LUBSK\_LANG\_MIN, ((LUBSK\_LANG\_MAX-LUBSK\_LANG\_MIN+XBUNIT)/XBUNIT)\*XBUNIT], 0  
  TES;  
|  
| The following masks are used to test OTSSSV\_OWNR to be sure that only  
| one bit is set.  
|  
LITERAL  
  OTSSSM\_OWNR\_BAS = 1 ^ LUBSK\_LANG\_BAS,  
  OTSSSM\_OWNR\_FOR = 1 ^ LUBSK\_LANG\_FOR;  
|     End of file OTSCCBREQ.REQ

M 12

15-Sep-1984 23:44:38  
15-Sep-1984 22:44:59

VAX-11 Bliss-32 V4.0-742  
S255\$DUA28:[FORRTL.SRC]FORLIB.REQ;1

Page 39 (1)

1298 0  
1299 0  
1300 0

REQUIRE 'RTLIN:OTSLNK';

! (Must come after OTSISB and OTSLUB)  
! Common linkage definitions

File: OTSLNK.REQ Edit: PLL1035

R1301 0  
R1302 0  
R1303 0  
R1304 0  
R1305 0  
R1306 0  
R1307 0  
R1308 0  
R1309 0  
R1310 0  
R1311 0  
R1312 0  
R1313 0  
R1314 0  
R1315 0  
R1316 0  
R1317 0  
R1318 0  
R1319 0  
R1320 0  
R1321 0  
R1322 0  
R1323 0  
R1324 0  
R1325 0  
R1326 0  
R1327 0  
R1328 0  
R1329 0  
R1330 0  
R1331 0  
R1332 0  
R1333 0  
R1334 0  
R1335 0  
R1336 0  
R1337 0  
R1338 0  
R1339 0  
R1340 0  
R1341 0  
R1342 0  
R1343 0  
R1344 0  
R1345 0  
R1346 0  
R1347 0  
R1348 0  
R1349 0  
R1350 0  
R1351 0  
R1352 0  
R1353 0  
R1354 0  
R1355 0  
R1356 0  
R1357 0

+ This file, OTSLNK.REQ, contains the definitions of all LINKAGE declarations  
for BLISS modules

-

\*\*\*\*\*

COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
ALL RIGHTS RESERVED.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

\*\*\*\*\*

Author: T. Hastings  
1-11 - Add CALL R0. TNH 29-July-78  
1-12 - Add JSB\_CCB\_GET. TNH 2-Aug-78  
1-13 - Change name to FORLNK.REQ. JBS 14-NOV-78  
1-014 - Add copyright notice. JBS 16-NOV-78  
1-015 - Change file name to OTSLNK.REQ. JBS 06-DEC-78  
1-016 - Add linkage for BMF (Basic major frame ptr) - R11. DGP 17-Dec-78  
1-017 - Fix some comments. JBS 18-DEC-78  
1-018 - Change JSB\_REC0 linkage to save registers needed for CHSFILL. DGP  
27-Feb-79  
1-019 - Similarly, change JSB\_UDFO, since the UDF routines must  
preserve the same registers to call the REC routines.  
JBS 29-FEB-1979  
1-020 - That change causes a similar change in JBS\_REC1 and JSB\_REC9.  
JBS 28-FEB-1979  
1-021 - Which in turn causes the same changes in JSB\_UDF9.  
JBS 28-FEB-1979  
1-022 - Which in turn causes the same changes in JBS\_DO\_READ and  
JSB DO WRITE. JBS 28-FEB-1979  
1-023 - Add linkage JSB\_CCB\_A1\_A0 for PUT relative with count. DGP 02-Mar-79  
1-024 - Add linkage JSB\_REC\_IND for indexed file support. DGP 03-Apr-79  
1-025 - Change linkage JSB\_REC\_IND to take 5 args. DGP 06-Apr-79  
1-026 - Add linkage for conversion kernel routine. DGP 27-Jun-79  
1-028 - Add linkage for Basic format routines. DGP 30-Jul-79  
1-029 - Remove PRINT statement, for new BLISS compiler. JBS 02-OCT-1979  
1-030 - Change JSB\_FORMAT A7 to A10. DGP 31-Oct-79  
1-031 - Add new linkage JSB\_REC\_WSL1. DGP 06-Nov-79

```

R1358 0 1-032 - Change JSB_UDFO, JSB_FMT0 so that they pass no arguments. SBL 5-Dec-1979
R1359 0 1-033 - Add linkage JSB_FMT1 for Fortran format interpreter. JAW
R1360 0 08-Aug-1981
R1361 0 1-034 - Change JSB_A5_R11 to JSB_A6_R11, JSB_A10_R11 to JSB_A11_R11.
R1362 0 PLL 16-Mar-1982
R1363 0 1-035 - Change JSB_DO_READ, JSB_REC_IND, JSB_REC0. PLL 1-Jun-1982
R1364 0 --
R1365 0
R1366 0 ++
R1367 0 Linkage definitions for BLISS modules for CALL and JSB routines
R1368 0 The idea is to have all definitions here in one place
R1369 0 so that they can be changed easily and the entire
R1370 0 R1L recompiled.
R1371 0 --
R1372 0
R1373 0 +
R1374 0 Define symbols for register numbers used to pass parameters from one
R1375 0 module to another. Note: these symbols are used in the modules in
R1376 0 GLOBAL REGISTER declarations rather than below in this REQUIRE file
R1377 0 (where all registers appear as absolute numbers).
R1378 0 -
R1379 0
R1380 0 LITERAL
R1381 0 K_BMF_REG = 11; ! Register used by Basic compiler to point
R1382 0 to last major frame
R1383 0 K_CCB_REG = 11; ! Pointer to LUB/ISB/RAB
R1384 0
R1385 0 +
R1386 0 First define some macros for frequently used combinations.
R1387 0 Do not change the PRESERVE conventions for our sanity.
R1388 0 Also do not change the definitions of these combination, since
R1389 0 they are also used in defining local routines within a module that is CALLED!!!
R1390 0 NOTE: Local routines which are JSBed to from JSB procedures must
R1391 0 have LINKAGE definitions here even though only local procedures.
R1392 0 Otherwise, lose control of NOTUSED registers which must be same
R1393 0 or more inclusive for JSB routines called by JSB routines.
R1394 0 See DO_READ and DO_WRITE for examples.
R1395 0 -
R1396 0
R1397 0 MACRO
R1398 0
R1399 0 +
R1400 0 CALL interface with CCB passed in R11 (in and/or out)
R1401 0
R1402 0
MR1403 0 CALL_CCB_R11 =
R1404 0 CALL: GLOBAL(CCB=11) %.
R1405 0
R1406 0
R1407 0 +
R1408 0 JSB interface with CCB passed in R11, 1 arg in R0 and 1 arg in R1 and
R1409 0 only uses R0 and R1
R1410 0
MR1411 0 JSB_CCB_A1_A0 =
R1412 0 JSB(Register = 1, REGISTER = 0): GLOBAL(CCB=11) NOTUSED(2,3,4,5,6,7,8,9,10) %.
R1413 0
R1414 0 +

```

.. R1415 0  
.. R1416 0  
.. R1417 0  
.. MR1418 0  
.. R1419 0  
.. R1420 0  
.. R1421 0  
.. R1422 0  
.. R1423 0  
.. R1424 0  
.. R1425 0  
.. MR1426 0  
.. R1427 0  
.. R1428 0  
.. R1429 0  
.. R1430 0  
.. R1431 0  
.. R1432 0  
.. MR1433 0  
.. R1434 0  
.. R1435 0  
.. R1436 0  
.. R1437 0  
.. R1438 0  
.. R1439 0  
.. MR1440 0  
.. R1441 0  
.. R1442 0  
.. R1443 0  
.. R1444 0  
.. R1445 0  
.. R1446 0  
.. MR1447 0  
.. R1448 0  
.. R1449 0  
.. R1450 0  
.. R1451 0  
.. R1452 0  
.. R1453 0  
.. R1454 0  
.. MR1455 0  
.. R1456 0  
.. R1457 0  
.. R1458 0  
.. R1459 0  
.. R1460 0  
.. R1461 0  
.. MR1462 0  
.. R1463 0  
.. R1464 0  
.. R1465 0  
.. R1466 0  
.. R1467 0  
.. R1468 0  
.. R1469 0  
.. R1470 0  
.. MR1471 0

! CALL interface with BMF passed in R11 (in and/or out)  
!-  
CALL\_BMF\_R11 =  
CALL: GLOBAL(BMF=11) %.

!+  
! CALL interface with 1st arg in R0  
!-  
CALL\_A0 =  
CALL (REGISTER = 0): %.

!+  
! JSB interface with CCB passed in R11, no args and uses R0-R5  
!-  
JSB\_CCB\_R5 =  
JSB: GLOBAL(CCB=11) NOTUSED (6,7,8,9,10) %.

!+  
! JSB interface with CCB passed in R11, no args and only uses R0, R1  
!-  
JSB\_CCB\_NO\_ARGS =  
JSB: GLOBAL(CCB=11) NOTUSED (2,3,4,5,6,7,8,9,10) %.

!+  
! JSB interface with CCB passed in R11, 1 arg in R0, and only uses R0, R1  
!-  
JSB\_CCB\_A0 =  
JSB (REGISTER = 0): GLOBAL(CCB=11) NOTUSED (2,3,4,5,6,7,8,9,10) %.

!+  
! JSB interface with CCB passed in R11, 1 arg in R0, and preserves  
through R5. Needed for MOVCS or calling routines that use MOVCS.  
!-  
JSB\_CCB\_A0\_R5 =  
JSB (REGISTER = 0): GLOBAL(CCB=11) NOTUSED (6,7,8,9,10) %.

!+  
! Same as above, but with 2 arguments.  
!-  
JSB\_CCB\_A1\_R5 =  
JSB (REGISTER = 0, REGISTER = 1): GLOBAL(CCB=11) NOTUSED (6,7,8,9,10) %.

!+  
! JSB interface with CCB passed in R11, 1 arg in R2, and only uses R0, R1, R2  
Needed when input arg is referenced after a CALL or JSB.  
so do not need to copy to R2.  
!-  
JSB\_CCB\_A2 =

R1472 0  
R1473 0  
R1474 0  
R1475 0  
R1476 0  
R1477 0  
R1478 0  
R1479 0  
MR1480 0  
R1481 0  
R1482 0  
R1483 0  
R1484 0  
R1485 0  
R1486 0  
R1487 0  
R1488 0  
MR1489 0  
MR1490 0  
R1491 0  
R1492 0  
R1493 0  
R1494 0  
R1495 0  
R1496 0  
R1497 0  
MR1498 0  
MR1499 0  
R1500 0  
R1501 0  
MR1502 0  
MR1503 0  
R1504 0  
R1505 0  
R1506 0  
R1507 0  
R1508 0  
R1509 0  
R1510 0  
R1511 0  
R1512 0  
MR1513 0  
R1514 0  
R1515 0  
R1516 0  
R1517 0  
R1518 0  
R1519 0  
MR1520 0  
R1521 0  
R1522 0  
P1523 0  
R1524 0  
R1525 0  
R1526 0  
R1527 0  
R1528 0

JSB (REGISTER = 2): GLOBAL(CCB=11) NOTUSED (3,4,5,6,7,8,9,10) %.

+ JSB interface with CCB passed in R11, 1 arg in R2, and preserves  
through R5. Needed in place of JBS\_CCB\_A2 to do MOVC5, or call  
routines which do.

-

JSB\_CCB\_A2 R5 =

+ JSB (REGISTER = 2): GLOBAL(CCB=11) NOTUSED (6,7,8,9,10) %.

+ JSB interface for Fortran format interpreter with CCB passed in  
R11, two arguments passed in R10 and P7, and routine value  
returned in R8.

-

JSB\_CCB\_FMT1 =

+ JSB: GLOBAL(CCB = 11, EL\_SIZE = 10, DT\_SEEN = 9, FMT\_CODE = 8)  
NOPRESERVE (2,3) NOTUSED (4,5,6,7) %.

+ Support for Indexed files.  
Pass arguments (6) in R0:R5 and CCB is passed in R11.

-

JSB\_CCB\_A6 R5 =

+ JSB (REGISTER = 0, REGISTER = 1, REGISTER = 2, REGISTER = 3,  
REGISTER = 4, REGISTER = 5):GLOBAL (CCB = 11) NOTUSED (6, 7, 8, 9, 10) %.

JSB\_CCB\_A5 R5 =

+ JSB (REGISTER = 0, REGISTER = 1, REGISTER = 2, REGISTER = 3,  
REGISTER = 4):GLOBAL (CCB = 11) NOTUSED (6, 7, 8, 9, 10) %.

+ JSB interface with CCB passed in R11, arg1 in R2, arg2 in R0,  
and only uses R0, R1, R2.  
Needed when input arg1 is referenced after a CALL or JSB, so save  
copying to R2.

-

JSB\_CCB\_A2 A0 =

+ JSB (REGISTER = 2, REGISTER = 0): GLOBAL(CCB=11) NOTUSED (3,4,5,6,7,8,9,10) %.

+ JSB interface (no CCB), args in R0 and R9

-

JSB\_A0 A1 R8 =

+ JSB (REGISTER = 0, REGISTER = 1) : NOPRESERVE (2,3,4,5,6,7,8) %.

+ JSB for Basic format routines - Plain F and E format. Pass  
6 args and preserve all other registers. (1 optional arg)

-

FC

```
MR1529 0
MR1530 0
R1531 0
R1532 0
R1533 0
R1534 0
R1535 0
R1536 0
R1537 0
MR1538 0
MR1539 0
MR1540 0
R1541 0
R1542 0
R1543 0
R1544 0
R1545 0
R1546 0
MR1547 0
R1548 0
R1549 0

JSB_A6_R11 =
JSB (REGISTER = 0, REGISTER = 1, REGISTER = 2, REGISTER = 3, REGISTER = 4, REGISTER = 5) :
PRESERVE (6, 7, 8, 9, 10, 11) %;

!+
JSB for Basic format routines - Fancy F and E formats. Pass 11 args and
preserve all other registers. (4 optional args)
!-
JSB_A11_R11 =
JSB (REGISTER = 0, REGISTER = 1, REGISTER = 2, REGISTER = 3, REGISTER = 4,
REGISTER = 5, REGISTER = 6, REGISTER = 7, REGISTER = 8, REGISTER = 9, REGISTER = 10) :
PRESERVE (11) %;

!+
JSB interface (without (CB), no args in registers
!-
JSB_NO_ARGS =
JSB: NOTUSED (2,3,4,5,6,7,8,9,10) %;
```

R1550 0  
R1551 0  
R1552 0  
R1553 0  
R1554 0  
R1555 0  
R1556 0  
R1557 0  
R1558 0  
R1559 0  
R1560 0  
R1561 0  
R1562 0  
R1563 0  
R1564 0  
R1565 0  
R1566 0  
R1567 0  
R1568 0  
R1569 0  
R1570 0  
R1571 0  
R1572 0  
R1573 0  
R1574 0  
R1575 0  
R1576 0  
R1577 0  
R1578 0  
R1579 0  
R1580 0  
R1581 0  
R1582 0  
R1583 0  
R1584 0  
R1585 0  
R1586 0  
R1587 0  
R1588 0  
R1589 0  
R1590 0  
R1591 0  
R1592 0  
R1593 0  
R1594 0  
R1595 0  
R1596 0  
R1597 0  
R1598 0  
R1599 0  
R1600 0  
R1601 0  
R1602 0  
R1603 0  
R1604 0  
R1605 0  
R1606 0

++  
Now define the LINKAGE declarations.  
Use names associated with the entry point rather than  
the type of linkage, so that we can easily change  
the linkage for an entry point without changing that  
for other entry points.  
Note: entry points that are dispatched to using a table  
must have the same linkage name. In this case the LINKAGE  
name is associated with the name of the dispatch table and  
the call is made using the general LINKAGE form.  
NOTUSED restriction!!! Because each JSB declaration must be  
aware of all JSB routines which are in turn called. The NOTUSED  
registers can only be the same as the caller (if also A JSB routine)  
or include additional registers as well as being the same.  
Thus, this file documents the calling tree for JSB linkages  
so that the NOTUSED declarations can be kept in agreement.  
--  
LINKAGE  
+  
Default CALL using CCB as a GLOBAL register,  
all args in arg list.  
-  
CALL\_CCB = CALL\_CCB\_R11.  
+  
CALL from BASIC compiled code, which uses R11 to point to the  
major frame.  
-  
CALL\_BMF = CALL\_BMF\_R11.  
+  
This is a linkage for BASS\$REC\_WSL1 to allow one arg to be passed.  
-  
JSB\_REC\_WSL1 = JSB\_CCB\_A0\_R5.  
+  
CALL passing first arg in R0.  
Used by FORENTRY module to make multiple entry points  
all branch to F1OBEG.  
-  
CALL\_F1OBEG = CALL\_A0.  
+  
UDF initialization (user data formatting level of abstraction)  
Arg is adr. of format statement.  
JSBs to record level initialization (JSB\_REC0).  
-  
JSB\_UDFO = JSB\_CCB\_R5.  
+  
++

6 13  
12-Sep-1984 23:44:38  
15-Sep-1984 23:02:30

VAX-11 Bliss-32 V4.0-742  
\_S255\$DUA28:[FORRTL.SRC]OTSLNK.REQ;1

Page 46  
(2)

R1607 0  
R1608 0  
R1609 0  
R1610 0  
R1611 0  
R1612 0  
R1613 0  
R1614 0  
R1615 0  
R1616 0  
R1617 0  
R1618 0  
R1619 0  
R1620 0  
R1621 0  
R1622 0  
R1623 0  
R1624 0  
R1625 0  
R1626 0  
R1627 0  
R1628 0  
R1629 0  
R1630 0  
R1631 0  
R1632 0  
R1633 0  
R1634 0  
R1635 0  
R1636 0  
R1637 0  
R1638 0  
R1639 0  
R1640 0  
R1641 0  
R1642 0  
R1643 0  
R1644 0  
R1645 0  
R1646 0  
R1647 0  
R1648 0  
R1649 0  
R1650 0  
R1651 0  
R1652 0  
R1653 0  
R1654 0  
R1655 0  
R1656 0  
R1657 0  
R1658 0  
R1659 0  
R1660 0  
R1661 0  
R1662 0  
R1663 0

JSB to plain formatting routines for Basic.
JSB\_FORMAT\_A6 = JSB\_A6\_R11,
+
JSB to fancy formatting routines for Basic.
-
JSB\_FORMAT\_A11 = JSB\_A11\_R11,
+
UDF termination (user data formatting level of abstraction)
JSBs to DO\_READ (JSB\_DO\_READ) or DO\_WRITE (JSB\_DO\_WRITE).
-
JSB\_UDF9 = JSB\_CCB\_R5,
+
UDF read routine
JSBs to record level (JSB\_REC1).
-
JSB\_DO\_READ = JSB\_CCB\_A1\_R5,
+
UDF write routine
JSBs to record level (JSB\_REC1).
-
JSB\_DO\_WRITE = JSB\_CCB\_A0\_R5,
+
Format interpreter initialization: FORMAT\_ADR = arg is adr. of format statement
JSBs to nothing.
-
JSB\_FMT0 = JSB\_CCB\_NO\_ARGS,
+
Format interpreter main processing
-
JSB\_FMT1 = JSB\_CCB\_FMT1,
+
JSB to REC level of index file support
-
JSB\_REC\_IND = JSB\_CCB\_A5\_R5,
JSB\_REC\_IND1 = JSB\_CCB\_A6\_R5.
+
Record Level RMS interface level initialization.
JSBs to nothing.

```
R1664 0 JSB_REC0 = JSB_CCB_R5.  
R1665 0 JSB_REC2 = JSB_CCB_A0_R5.  
R1666 0  
R1667 0  
R1668 0 |+ Record level RMS interface level finished one buffer  
R1669 0 | JSBs to nothing.  
R1670 0 |  
R1671 0 JSB_REC1 = JSB_CCB_R5.  
R1672 0  
R1673 0  
R1674 0  
R1675 0 |+ Record level RMS interface termination of statement.  
R1676 0 | JSBs to nothing.  
R1677 0 |  
R1678 0  
R1679 0 JSB_REC9 = JSB_CCB_R5.  
R1680 0  
R1681 0  
R1682 0 |+ Push current LUB/ISB/RAB: LOGICAL_UNIT is unit no., LUN_MIN is min. no.  
R1683 0 | JSBs to nothing.  
R1684 0 |  
R1685 0  
R1686 0 JSB_CB_PUSH = JSB_CCB_A2_A0.  
R1687 0  
R1688 0  
R1689 0 |+ PUT relative with count  
R1690 0 |  
R1691 0  
R1692 0 JSB_PUT = JSB_CCB_A1_A0.  
R1693 0  
R1694 0  
R1695 0 |+ Pop current LUB/ISB/RAB  
R1696 0 | JSBs to nothing.  
R1697 0 |  
R1698 0  
R1699 0 JSB_CB_POP = JSB_CCB_NO_ARGS.  
R1700 0  
R1701 0  
R1702 0 |+ Return current LUB/ISB/RAB to free storage (open error or close)  
R1703 0 | JSBs to nothing.  
R1704 0 |  
R1705 0  
R1706 0 JSB_CB_RET = JSB_CCB_NO_ARGS.  
R1707 0  
R1708 0  
R1709 0 |+ Get adr. of current LIB/ISB/RAB (called only from non-shared routines)  
R1710 0 | since harder to have a data entry vector which is the same  
R1711 0 | if module were to become shared or vice versa.  
R1712 0 |  
R1713 0  
R1714 0 JSB_CB_GET = JSB_CCB_NO_ARGS.  
R1715 0  
R1716 0  
R1717 0 |+ JSB to kernel conversion routine  
R1718 0 |  
R1719 0  
R1720 0 JSB_CVT_KERNEL = JSB_A0_A1_R8.
```

R1721 0  
R1722 0  
R1723 0  
R1724 0  
R1725 0  
R1726 0  
R1727 0  
R1728 0

+ JSB to CALL\_VFE routine, args on stack
JSB\_CALL\_VFE = JSB\_NO\_ARGS;  
! End of file OTSLNK.REQ

15-Sep-1984 23:44:38  
15-Sep-1984 22:44:59

VAX-11 Bliss-32 V4.0-742  
S255\$DUA28:[FORRTL.SRC]FORLIB.REQ;1 Page 49 (1)

: 1729 0  
: 1730 0 REQUIRE 'RTLIN:OTSMAC';

! Common macros

R1731 0  
R1732 0  
R1733 0  
R1734 0  
R1735 0  
R1736 0  
R1737 0  
R1738 0  
R1739 0  
R1740 0  
R1741 0  
R1742 0  
R1743 0  
R1744 0  
R1745 0  
R1746 0  
R1747 0  
R1748 0  
R1749 0  
R1750 0  
R1751 0  
R1752 0  
R1753 0  
R1754 0  
R1755 0  
R1756 0  
R1757 0  
R1758 0  
R1759 0  
R1760 0  
R1761 0  
R1762 0  
R1763 0  
R1764 0  
R1765 0  
R1766 0  
R1767 0  
R1768 0  
R1769 0  
R1770 0  
R1771 0  
R1772 0  
R1773 0  
R1774 0  
R1775 0  
R1776 0  
R1777 0  
R1778 0  
R1779 0  
R1780 0  
R1781 0  
R1782 0  
R1783 0  
R1784 0  
R1785 0  
R1786 0  
R1787 0

♦ This file, OTSMAC.REQ, defines OTS macros.  
Edit: SBL1039

\*\*\*\*\*

♦ COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
ALL RIGHTS RESERVED.

♦ THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

♦ THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

♦ DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

\*\*\*\*\*

♦ Author: T. Hastings

1-25 - REQUIRE LPSECT. TNH 19-Dec-77  
1-26 - Remove SET CB BASE(). JMT 12-Apr-78  
1-27 - Use RTLIN: Logical name in REQUIRE. TNH 28-Apr-78  
1-28 - Define ADR\_VECTOR. TNH 7-June-78  
1-30 - Change name to FORMAC.REQ (with apologies to Dick Gruen)  
and change name of LPSECT to RTLSECT JBS 14-NOV-78  
1-031 - Add a copyright notice JBS 16-NOV-78  
1-032 - Change file name to OTSMAC.REQ and remove REQUIRE of RTLSECT.  
(Let users of OTSMAC.REQ also REQUIRE RTLSECT.) JBS 06-DEC-78  
1-033 - Add offsets and lengths of the dispatch tables. JBS 25-JUN-1979  
1-034 - Make them weak globals so they can be used by macro routines.  
JBS 26-JUN-1979  
1-035 - Remove FORTRAN offsets and lengths (moved to ISB). JBS for SBL  
12-JUL-1979  
1-036 - Remove BASIC offsets and lengths (moved to ISB). JBS 12-JUL-1979  
1-037 - Remove PRINT statement, for new BLISS compiler. JBS 02-OCT-1979  
1-038 - Add COPY\_BYT\_A, COPY\_WORD\_A, COPY\_LONG\_A, COPY\_QUAD\_A  
macros. SBL 18-Dec-1979  
1-039 - Add ONE\_OF macro. SBL 18-Dec-1981

--

♦ Macro for writing a character string and then advancing pointer  
Designed so that it is placed on the left side of a substitution  
statement. Anticipates feature being added to BLISS as a form  
for (HSWCHAR\_A (DESPA) if looks good.

```
R1788 0
R1789 0
R1790 0
R1791 0
R1792 0
MR1793 0
MR1794 0
MR1795 0
MR1796 0
MR1797 0
R1798 0
R1799 0
R1800 0
R1801 0
R1802 0
R1803 0
R1804 0
R1805 0
R1806 0
R1807 0
R1808 0
R1809 0
R1810 0
MR1811 0
MR1812 0
R1813 0
R1814 0
R1815 0
R1816 0
R1817 0
R1818 0
R1819 0
R1820 0
R1821 0
R1822 0
R1823 0
MR1824 0
R1825 0
R1826 0
R1827 0
R1828 0
R1829 0
R1830 0
R1831 0
R1832 0
R1833 0
R1834 0
R1835 0
R1836 0
R1837 0
R1838 0
R1839 0
R1840 0
R1841 0
R1842 0
PR1843 0
PR1844 0

; Call: CH_WCHAR_A (CS_POINTER_ADR.ma.r) = ... ;
;-
MACRO
  CH_WCHAR_A (CS_POINTER_ADR) =
  (LOCAL T;
  T = .CS_POINTER_ADR;
  CS_POINTER_ADR = CH$PLUS (.CS_POINTER_ADR, 1);
  .T)<0,8> %;

;*
; Macro for writing a character without advancing the pointer.
; Desinged so that is placed on the left of a substitution statement.
; Anticipates feature being added to BLISS as a form
; for CH$WCHAR (DSTPV) if looks good.
; Call: CH_WCHAR (CS_POINTER.ra.v) = ... ;
;-
MACRO
  CH_WCHAR (CS_POINTER_VAL) =
  (CS_POINTER_VAL)<0,8> %;

;*
; Macros for processing the compiled format text byte strings.
;-
MACRO
  RBYTE_A(P) = (P = .P+1; .(P-1)<0, 8>) %,
  RWORD_A(P) = (P = .P+2; .(P-2)<0,16>) %,
  RLONG_A(P) = (P = .P+4; .(P-4)<0,32>) %,
  CALL_VFE(P)=
  ? (LOCAL T; T = .(P)<0,32>; P = .P+4; .T+.P) () ) %;

;*
; Macros for copying values referenced by pointers.
;-
MACRO
  COPY_BYTE_A (S,D) = (D=.D+1; (.D-1)<0,8>=RBYTE_A(S)) %,
  COPY_WORD_A (S,D) = (D=.D+2; (.D-2)<0,16>=RWORD_A(S)) %,
  COPY_LONG_A (S,D) = (D=.D+4; (.D-4)<0,32>=RLONG_A(S)) %,
  COPY_QUAD_A (S,D) = ((.D)<0,32>=.(S)<0,32>; (.D+4)<0,32>=.(S+4)<0,32>; D=.D+8; S=.S+8) %;

;*
; Macro to complete the transportable character pointer notion.
; Everywhere that an address (A) can be specified in BLISS,
; allow a character pointer with mnemonic P (rather than CP to keep one letter)
;-
MACRO
  ZBLISS32 (
```

```
PR1845 0      LSSP = LSSA %;  
PR1846 0      LEQP = LEQA %;  
PR1847 0      EQLP = EQLA %;  
PR1848 0      NEQP = NEQA %;  
PR1849 0      GEQP = GEQA %;  
PR1850 0      GTRP = GTRA %;  
PR1851 0      MAXP = MAXA %;  
R1852 0      MINP = MINA %;);  
R1853 0  
R1854 0  
R1855 0      /* Clear a vector of BLISS values (transportable)  
R1856 0      */  
R1857 0  
R1858 0      MACRO  
MR1859 0      FILL VAL (VALUE, LENGTH, ADDRESS) =  
R1860 0      %BLISS32 (CH$FILL (VALUE, (LENGTH) + %UPVAL, ADDRESS)) %;  
R1861 0  
R1862 0  
R1863 0  
R1864 0      /* Allocate string descriptor  
R1865 0      Rest of descriptor symbols are defined in SRMDEF.MDL  
R1866 0      But currently no way in MDL to define a macro  
R1867 0  
R1868 0      To declare and allocate a descriptor:  
R1869 0  
R1870 0      LOCAL  
R1871 0      name: DSC$descriptor;  
R1872 0      */  
R1873 0  
R1874 0  
R1875 0      MACRO  
DSC$descriptor = BLOCK[8, BYTE] %;      ! MDL requires BYTE
```

```

R1876 0
R1877 0
R1878 0
R1879 0
R1880 0
R1881 0
R1882 0
R1883 0
R1884 0
R1885 0
R1886 0
R1887 0
R1888 0
R1889 0
R1890 0
R1891 0
R1892 0
R1893 0
R1894 0
MR1895 0
MR1896 0
MR1897 0
R1898 0
R1899 0
R1900 0
MR1901 0
R1902 0
R1903 0
MR1904 0
MR1905 0
MR1906 0
MR1907 0
MR1908 0
MR1909 0
R1910 0
R1911 0
MR1912 0
MR1913 0
MR1914 0
MR1915 0
MR1916 0
MR1917 0
MR1918 0
R1919 0
R1920 0
R1921 0
R1922 0
R1923 0

```

! MACRO

### THE "ONE OF" MACRO

Macros to determine if the value of an expression is one of a set of specified small-integer values. These macros can be used only if the following conditions are met:

The value to be tested is in the range 0 through 127.

The values to be tested for are all in the range 0 through 31.

Example:

IF ONE\_OF (.X, 1,3,5) ...

The code generated is much more efficient than a series of comparisons (provided that the values being tested are all compile-time constants).

XBMSK [A]=

```

  XIF NOT XCTCE(A) XTHEN XERRORMACRO('ONE_OF argument not a CTCE') XFI
  XIF (A GTRU 31) XTHEN XERRORMACRO('ONE_OF constant greater than 31') XFI
  (1 ^ (31 - (A))) %,
```

BMSK []=

```

  TO OR XBMSK_(%REMAINING) %,
```

XCMP [A,B,C]=

```

  XIF %LENGTH EQL 3
  XTHEN
    ((A EQLU B) OR (A EQLU C))
  XELSE
    (A EQLU B)
  XFI %,
```

ONE\_OF(A)=

```

  XIF %LENGTH LEQ 1 XTHEN XERRORMACRO('Too few arguments to ONE_OF') XFI
  XIF %LENGTH LEQ 3
  XTHEN
    XCMP_(A,%REMAINING)
  XELSE
    (( ( BMSK_(%REMAINING) ) ^ (A) ) LSS 0)
  XFI %;
```

! End of file OTSMAC.REQ

B 14  
15-Sep-1984 23:44:38  
15-Sep-1984 22:44:59

VAX-11 Bliss-32 V4.0-742  
\$\_255\$DUA28:[FORRTL.SRC]FORLIB.REQ;1 Page 54 (1)

:: 1924 0  
:: 1925 0 ! End of file FORLIB.REQ

Library Statistics

File	-----	Symbols	-----	Pages	Processing
	Total	Loaded	Percent	Mapped	Time
\$_255\$DUA28:[SYSLIB]STARLET.L32;1	9776	5	0	581	00:01.0

COMMAND QUALIFIERS

BLISS/LIBRARY=LIB\$:/LIST=LIS\$:/SOURCE=REQUIRE SRC\$:FORLIB

Run Time: 00:19.9  
Elapsed Time: 01:19.3  
Lines/CPU Min: 5818  
Lexemes/CPU-Min: 26644  
Memory Used: 164 pages  
Library Precompilation Complete

0181 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY

